Implantación de un sistema de certificados

César Guzmán de Lapuente

21 de junio de 2011

Índice general

Índice de figuras

Índice de cuadros

Capítulo 1

Conceptos básicos

1.1. Introducción

La fortaleza de la criptografía de clave secreta moderna se deriva de mantener secretas las claves, no los algoritmos. Esta característica es más una imposición que una elección. Los algoritmos deben ser públicos para soportar un análisis técnico profundo. En ausencia de este escrutinio detallado, el algoritmo puede ocultar sus debilidades y, además, tarde o temprano alguien, mediante un proceso de ingeniería inversa, podrá averiguar el algoritmo.

Las claves secretas deben distribuirse entre las partes que se desean comunicar y su transporte requiere el establecimiento de canales seguros. El concepto de criptografía de clave pública ofrece un mecanismo flexible y escalable que permite resolver el problema de la distribución de claves secretas. En la criptografía de clave pública, cada usuario dispone de dos claves: la clave pública y la clave privada. La clave privada sólo debe ser conocida por el usuario que la generó, mientras que la clave pública se distribuye libremente. Una de las premisas es que sea computacionalmente inviable calcular la clave privada a partir de la pública. Además, la criptografía de clave pública ofrece varios servicios adicionales: firmas digitales, no-repudio, integridad de los datos,.... El requisito de la distribución libre de la clave pública tiene un coste que consiste en confiar en que la clave pública se ha asociado con su legítimo usuario. Una solución contrastada consiste en el uso de los certificados y, adicionalmente, de toda la infraestructura que los soporta.

Una *Infraestructura de Clave Pública* (Public Key Infrastructure, PKI) es una infraestructura que proporciona servicios de seguridad a las aplicaciones informáticas y sus usuarios mediante el uso de técnicas de clave pública de

un modo transparente al usuario. Entre los beneficios más destacados que proporciona una PKI se incluyen el ahorro de costes y la interoperabilidad de soluciones.

Este estudio pretende analizar los distintos aspectos que se deben considerar al implantar un sistema de certificados (PKI). Para ello nos basaremos en los pasos que una empresa u organización debe dar para implantar una PKI.

Al igual que cualquier otro proyecto de tecnologías de la información (IT) llevado a cabo por una empresa u organización, la implantación de un sistema de certificados puede subdividirse en 4 etapas claves:

- Evaluación
- Diseño
- Despliegue
- Mantenimiento

Debe quedar claro que el objetivo del proyecto no es implantar un sistema de certificados en una organización, sino analizar los distintos puntos que se deben considerar al implantar una PKI. Las 4 etapas claves de un proyecto de IT sirven simplemente como un armazón en el cual se encajan los distintos aspectos a estudiar. Puesto que se trata de un proyecto teórico, no centrado en ninguna organización o empresa real, se darán ideas y pautas generales.

Debido a todo lo dicho anteriormente, el actual estudio seguirá la siguiente estructura:

En el primer capítulo, tras una breve introducción histórica, se explicarán los diferentes componentes que constituyen una PKI y las funciones que realizan.

El segundo capítulo se ocupa de la fase de evaluación. Se estudiarán los beneficios que produce una PKI y las motivaciones que inducen a una organización a su implantación, se comentará la utilidad de los documentos CP y CPS, se presentará un estudio de los costes que se derivan de la implantación y posterior mantenimiento de la PKI, se presentarán criterios para decidir si se externaliza o no el servicio y la elección del proveedor o fabricante.

El tercer capítulo trata del diseño tecnológico y despliegue de la solución PKI. Se considerarán cuestiones relativas a las diferentes arquitecturas PKI existentes, el diseño de certificados, los parámetros que se deben configurar en un servidor, cuestiones relativas a interoperabilidad, así como cuestiones

relativas a su integración con el entorno IT existente y el desarrollo de aplicaciones o módulos PKI a medida.

El capítulo dedicado al mantenimiento detallará todas las operaciones que una PKI completa debe llevar a cabo de una forma rutinaria. Estas actividades constituyen lo que se denomina *ciclo de vida de certificados y claves*. Además se describirán los protocolos que se han definido para llevar a cabo dichas operaciones. Por último, este capítulo incluirá algunas consideraciones sobre temas como soporte a usuarios, formación de usuarios y administradores, backups y prevención de desastres.

El último capítulo analizará un ejemplo concreto de PKI: el Documento Nacional de Identidad electrónico (DNI-e). Además, examinará las tendencias actuales en el desarrollo de las PKIs y realizará algunas consideraciones sobre su viabilidad futura.

1.2. Historia de las PKI

Oficialmente la criptografía de clave pública (Public Key Cryptography, PKC) comienza en 1976 cuando Whitfield Diffie y Martin E. Hellman, de la Universidad de Stanford, publican un artículo visionario titulado *New Directions in Cryptography*. Paralelamente, pero de forma independiente, Ralph C. Merkle propuso ideas similares. Sin embargo, hay una cierta controversia. De acuerdo con documentos publicados por el gobierno británico en 1997, la PKC fue originariamente inventada en el GCHQ (Government Communications Headquarters). El GCHQ es la organización secreta británica que se creó durante la Segunda Guerra Mundial y que, entre otros éxitos, consiguió descifrar el sistema ENIGMA usado por los nazis. Brevemente, Jim Ellis tuvo la idea de la criptografía de clave pública en 1970. En 1973 Clifford Cocks inventó una variante de RSA y unos meses más tarde Malcolm Williamson inventó un sistema similar al intercambio de claves de Diffie-Hellman. Sin embargo, los documentos publicados por el gobierno británico no muestran ninguna referencia a las firmas digitales. La historia no acaba ahí, pues parece que la estadounidense National Security Agency (NSA) también había hecho el mismo descubrimiento unos años antes. Si se desea encontrar una explicación algo más detallada sobre el descubrimiento de la criptografía de clave pública por parte de los servicios secretos británicos puede ser útil consultar el enlace *http: //cryptome. org/ukpk - alt.htm*.

Volviendo a la historia oficial, Diffie y Hellman vaticinan en su artículo que está a punto de producirse una revolución en el campo de la criptografía debido a que el desarrollo de hardware barato la libera de costosos dispositivos mecánicos y, a su vez, permite su aplicación en simples terminales de ordenadores o cajeros automáticos. En dicho artículo también proponen un sistema que minimiza la necesidad de canales seguros para la distribución de claves y prevén la posibilidad del equivalente a una firma escrita. Durante milenios se había admitido unánimemente que para que dos partes establecieran una comunicación segura debían previamente intercambiar una clave secreta de algún tipo; el artículo de Diffie y Hellman acaba con esa creencia.

El algoritmo de intercambio de claves propuesto por Diffie, Hellman y Merkle proporcionaba un mecanismo seguro para la distribución de claves, pero no implementaba firmas digitales. Tras leer el artículo de Diffie y Hellman, tres investigadores del MIT, Ronald Rivest, Adi Shamir y Leonard Adleman (RSA), comenzaron la búsqueda de una función matemática que implementara un sistema PKC completo. Finalmente descubrieron un algoritmo elegante basado en la multiplicación de dos números primos que encajaba exactamente en los requisitos para implementar un sistema PKC. Acababa de nacer el algoritmo RSA. RSA es el sistema de clave pública más ampliamente usado. Puede usarse para proporcionar confidencialidad y firmas digitales y su seguridad se basa en la dificultad de factorizar enteros.

En 1978, Loren M. Kohnfelder en su tesis de licenciatura en el MIT titulada *Towards a practical Public Key Cryptosystem* dirigida por Len Adleman propone el concepto de *certificado* para simplificar la implementación de RSA y soslayar algunos problemas relacionados. A partir de ese momento, los certificados y los sistemas de clave pública se desarrollaron en paralelo. Otro hito histórico destacado tuvo lugar en 1985 cuando Victor S. Miller y Neal Koblitz sugieren el uso de las curvas elípticas en la PKC.

Buena parte del interés en el desarrollo de las PKIs es el resultado del crecimiento de Internet. Concretamente, el aumento del comercio electrónico ha provocado una toma de conciencia relativa a las cuestiones de seguridad lo que, a su vez, ha incrementado el esfuerzo dedicado a estandarizar todos los aspectos de las PKIs. En 1988 se publica la primera versión de la recomendación X.509 que proporciona un formato normalizado para certificados y CRLs (*Certificate Revocation List*). Periódicamente han ido apareciendo nuevas versiones del estándar para incorporar nuevas funcionalidades. La IETF, responsable de los estándares que gobiernan la operación de Internet, ha escri-

to una serie de RFCs para el uso de una PKI en Internet basada en el estándar X.509. Tal conjunto de documentos se engloban bajo el nombre de PKIX. La misma IETF ha propuesto otro esquema de certificado más simple que el X.509 denominado SPKI, pero que no ha tenido una amplio seguimiento. El grupo ISO TC68 ha adaptado los certificados X.509 al sector financiero. ANSI X9F ha publicado un significativo número de estándares también relativos al sector bancario. Otras comunidades, si bien no definen estándares, especifican la implementación de productos a un cierto nivel, como por ejemplo el *U.S. Federal Public-Key Infrastrucute* o el *Government of Canada Public-Key Infrastructure*. Cabe decir que en el campo de los estándares se ha progresado significativamente y se ha alcanzado una madurez que permite una sólida implementación, despliegue e interoperación.

La evolución comercial de las PKIs es algo más irregular. A la publicación del artículo de Diffie y Hellman, le siguió un periodo de gran expectación que provocó una sobrevaloración de lo que la nueva tecnología podía aportar. Esta sobrevaloración contribuyó a un posterior desencanto auspiciado también por el coste de implementación de la tecnología, la falta de personal especializado y la incomprensión de los gestores. La consecuencia fue el abandono de muchos proyectos de implantación de PKIs. Este período de crisis se refleja en la desaparición de la empresa Baltimore Technologies que pocos años antes había visto como su cotización en bolsa se disparaba debido a que su área de negocio en el campo de los certificados digitales era vital para el comercio electrónico. A pesar de todo, algunas organizaciones persistieron en el uso de la tecnología hasta que consiguieron que cumpliera sus requisitos. Poco a poco la tecnología se ha ido asentando y su implementación se ha ido simplificando de modo que ha recuperado la confianza de la mayoría.

Sin embargo, las PKIs también tienen sus detractores y entre ellos destaca Bruce Schneier. En un trabajo escrito junto con Carl Ellison titulado *Ten risks of PKI: What you're not been told about Public Key Infrastructure* critica la tecnología y refuta el argumento de la necesidad de las PKIs para el florecimiento del comercio electrónico. Sin embargo, a medida que mejora la comprensión de la tecnología se van rebatiendo los puntos de vista de Bruce Schneier. Hay un cierto consenso en la actualidad sobre el hecho de que los beneficios de una PKI compensan el coste de su mantenimiento y también sobre el hecho de que las PKIs abren un horizonte de nuevas posibilidades de negocio que antes eran inviables por riesgos de seguridad, legislación, etc.

1.3. Certificados

En pocas palabras, los certificados de clave pública son estructuras de datos firmadas que se usan para asociar el nombre de una entidad (y otros atributos adicionales relacionados con la entidad) con la correspondiente clave pública.

A lo largo del tiempo se han estandarizado diferentes formatos de certificados de clave pública:

- Certificados de clave pública X.509
- Certificados SPKI (Simple Public Key Infrastructure)
- Certificados PGP (Pretty Good Privacy)
- Certificados CV (Card Verifiable)

Entre todos ellos, los certificados de clave pública X.509 son los que han sido más ampliamente adoptados. Estos certificados han ido evolucionando con el tiempo originando diferentes versiones. Los certificados de clave pública de la versión 1 son un subconjunto de los certificados de la versión 2 y éstos a su vez son un subconjunto de los de la versión 3. Debido a que la versión 3 incluye numerosas extensiones opcionales, estos certificados se han instanciado a su vez para adaptarse a aplicaciones específicas dando lugar, por ejemplo, a los certificados PKIX que se usan en Internet y a los certificados SET usados para pagos con tarjetas de crédito. La práctica totalidad de las PKIs usa certificados de clave pública X.509 v3 (versión 3) y es por ello que este trabajo se centra en dicho tipo de certificados.

Originalmente el documento X.509 especificaba los mecanismos de autenticación para el directorio X.500. El directorio X.500 requería mecanismos fuertes de autenticación para asegurar que sólo usuarios autorizados podían modificar o acceder a sus datos. La primera versión del documento publicada en 1988 contenía la versión 1 de los certificados X.509. Con el tiempo, el objeto de atención del documento X.509 dejó de ser el directorio (que nunca ha llegado a implementarse) para pasar a ser una PKI de propósito general. La versión 1 de los certificados presentaba problemas relativos a la renovación de certificados de CAs y una notable inflexibilidad para soportar atributos adicionales, por lo que en 1993 apareció la versión 2 que agregaba dos campos al formato antiguo. Los intentos de usar este nuevo formato en una PKI para el

Internet Privacy Enhanced Mail resultaron infructuosos y revelaron deficiencias. En respuesta a los nuevos requisitos, la ISO junto con IEC e ITU desarrollaron la versión 3 de los certificados X.509 en 1997. Esta versión incluye el poderoso mecanismo de las extensiones que, de modo flexible, permite incluir en los certificados información no soportada en los campos básicos. Las nuevas versiones del documento X.509 simplemente han añadido nuevas extensiones y han especificado los *Certificados de Atributos* sobre los que se construye las Privilege Management Infrastructures (PMI), pero no han alterado las ideas de la versión 3. La última versión del documento X.509 es de Noviembre del 2008.

La IETF (Internet Engineering Task Force) ha adaptado los certificados X.509 v3 para su uso en Internet promoviendo el uso de algunas extensiones y desaconsejando el uso de otras, entre ellas las que dieron lugar a los certificados X.509 v2. La IETF ha generado las recomendaciones PKIX para adaptar una PKI al entorno de Internet. De hecho, aunque los estándares PKIX están dirigidos principalmente a la comunidad de Internet, algunas de sus recomendaciones pueden aplicarse al entorno de la empresa y mantener de este modo la consistencia.

1.3.1. Estructura y semántica de los certificados

A continuación se van a detallar los diferentes campos que constituyen los certificados X.509 v3. La figura 1.1 representa el contenido de un certificado.

- *Version*. Este campo indica la versión a la cual se ajusta el formato del certificado. Puede referirse a la versión 1 (valor 0), versión 2 (valor 1) o versión 3 (valor 2).

- *Serial Number*. Es un entero asignado por la CA al certificado en el momento de su emisión y debe ser único dentro del ámbito de cada CA.

- *Signature Algorithm*. Este campo identifica el algoritmo criptográfico usado por la autoridad emisora al firmar el certificado. Para ello especifica el OID (Object Identifier) del algoritmo y sus parámetros asociados. Uno de los más usados hasta la fecha es SHA-1 con cifrado RSA. Este campo aparece dos veces en el certificado (dentro y fuera de la porción firmada). Los valores de los dos campos deben coincidir.

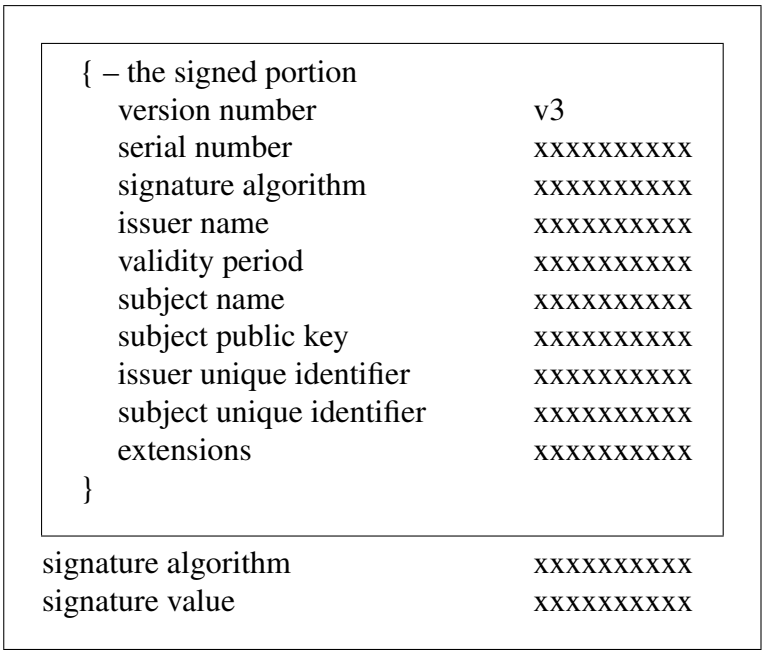

Figura 1.1: Elementos de un certificado X.509 v3

- *Issuer Name*. Es el Nombre Distinguido (Distinguished Name, DN) de la CA que emitió el certificado y debe estar siempre presente. Este campo es común a todos los certificados emitidos por la misma CA. La combinación de Serial Number e Issuer Name identifican unívocamente un certificado.

- *Validity Period*. Indica la ventana de tiempo en la que el certificado es considerado válido salvo que haya sido revocado. Se representa como una secuencia de 2 fechas. La primera componente de la fecha marca el inicio del periodo de validez, mientras que la segunda marca el final del periodo de validez.

- *Subject Name*. Indica el DN del propietario del certificado y puede ser nulo en el caso de que se use un formato alternativo de nombre en las extensiones.

- *Subject Public Key*. Este campo es una secuencia de 2 campos. Uno de

ellos representa el valor real de la clave pública certificada, el otro es el OID del algoritmo asociado a la clave.

- *Issuer Unique Identifier*. Es un campo opcional que identifica universalmente la autoridad que firma el certificado. Se usa sólo en las versiones 2 y 3. Su uso no está recomendado.

- *Subject Unique Identifier*. Es un campo opcional que identifica el sujeto del certificado. Se usa sólo en las versiones 2 y 3. Su uso no está recomendado.

Las extensiones son un mecanismo disponible en los certificados de versión 3 que añade flexibilidad y que permite confeccionar diferentes mecanismos de seguridad y protocolos. Cada extensión del certificado se asocia con un flag de criticidad y, casi siempre, son los propios estándares los que imponen la criticidad de las diversas extensiones. Una extensión que se ha marcado como crítica debe ser procesada y comprendida cuando se valida el certificado, de lo contrario se descarta el certificado. Una extensión marcada como no crítica que no es comprendida cuando se valida el certificado se ignora y se procede como si no estuviera presente. En cambio, si se comprende la extensión no crítica debe ser procesada del mismo modo que si se hubiera marcado como crítica. A continuación se pasa a describir las extensiones más importantes.

- *Authority Key Identifier*. Es un identificador único de la clave que se debe utilizar para verificar la firma digital del certificado. Distingue entre las múltiples claves del mismo emisor de certificados.

- *Subject Key Identifier*. Es un identificador único asociado con la clave pública contenida en el certificado. Distingue entre las múltiples claves del mismo propietario de certificados.

- *Key Usage*. Es una cadena de bits usada para identificar las funciones que soporta la clave pública del certificado. Por ejemplo: firmas digitales, cifrado de claves, cifrado de datos, firma de certificados, firma de CRLs,...

- *Extended Key Usage*. Es una secuencia de OIDs que identifican usos específicos de la clave pública certificada. Por ejemplo: protección de

e-mail, firma de código, autenticación de servidor TLS, autenticación cliente TLS,...

- *CRL Distribution Point*. Indica la ubicación donde reside la CRL asociada al certificado.

- *Private Key Usage Period*. Indica la ventana de tiempo en la que la clave privada correspondiente a la clave pública del certificado puede ser usada. Está pensado para claves de firmas digitales. El uso juicioso de esta extensión puede proporcionar un margen de tiempo entre el instante en el que la clave privada expira y el instante en que el certificado expira. Esto ayuda a eliminar casos en los que firmas digitales válidas son cuestionadas debido a que los periodos de vida de las claves son muy próximos o incluso idénticos.

- *Certificate Policies*. Indican una secuencia de uno o más OIDs de políticas asociadas con la emisión y uso subsiguiente del certificado. Si esta extensión se marca crítica, la aplicación debe adherirse al menos a una de las políticas indicadas o el certificado es descartado.

- *Policy Mappings*. Indica la equivalencia entre OIDs de políticas definidos en dos dominios de CAs. Sólo están presentes en certificados de CAs.

- *Subject Alternative Name*. Indica nombres alternativos del poseedor del certificado. Los formatos comúnmente usados de nombres alternativos son direcciones de e-mail, direcciones IP, nombres de servicios DNS,...

- *Issuer Alternative Name*. Indica nombres alternativos asociados al emisor del certificado. Algunos formatos usados son, como en la extensión anterior, direcciones de e-mail o direcciones IP.

- *Basic Constraints*. Permite distinguir el certificado de una CA del de una entidad final. Debe aparecer en todos los certificados de CAs para afirmar que se trata de un certificado de CA. Adicionalmente esta extensión puede contener un campo opcional para indicar el máximo número de certificados de CAs que pueden seguir a este certificado en un camino de certificación.

- *Name Constraints*. Una extensión sólo presente en certificados de CA. El propósito de esta extensión es restringir el espacio de nombres de los sujetos que emanan del certificado de CA y es aplicable tanto al campo Subject Name como a la extensión Subject Alternative Name.
- *Policy Constraints*. Esta extensión está sólo presente en certificados de CAs y puede usarse para prohibir Policy Mappings o para requerir que cada certificado en el camino de validación tenga unos OIDs de políticas determinados.
- *Inhibit Any Policy*. Esta extensión está sólo presente en certificados de CAs e indica que el OID correspondiente a *Any Policy* no debe ser considerado como un identificador de políticas.
- *Freshest CRL Pointer*. Proporciona un puntero a la información de CRL más reciente. En la práctica se trata de un puntero a una Delta CRL.

Según el estándar X.509 se pueden introducir extensions privadas para campos de aplicación específicos. El grupo de trabajo del PKIX ha introducido extensiones privadas para su uso en Internet. A continuación se detallan algunas de ellas.

- *Authority Information Access*. Especifica cómo se puede obtener información o servicios ofrecidos por el emisor del certificado. Se usa, por ejemplo, para servicios de validación on-line basados en OCSP.
- *Subject Information Access*. Especifica cómo se puede obtener información o servicios ofrecidos por el sujeto del certificado. Se usa, por ejemplo, para identificar la ubicación del repositorio donde la CA publica información de certificado y CRL.

Cualquier cambio de la información contenida en un certificado antes de que expire obliga a que el certificado sea revocado y se emita un nuevo certificado. Por tanto, debe procurarse que la información de los certificados sea estática para evitar una innecesaria revocación y reemisión de certificados.

1.4. Revocación de certificados

En ocasiones puede ser necesario deshacer la asociación entre entidad y clave pública que establece un certificado antes de que expire. En tal caso

se procede a revocar el certificado. Las circunstancias que obligan a revocar un certificado son muy variadas y, entre ellas, se pueden citar el compromiso de la clave privada, compromiso de la CA, el empleado abandona la compañía,... Las formas más comunes de implementar la revocación de certificados son mecanismos de publicación periódicos, tales como *Listas de Revocación de Certificados* (Certificate Revocation Lists, CRLs), o mecanismos de consulta on-line tales como el *Online Certificate Status Protocol* (OCSP). A continuación se exploran ambas soluciones.

1.4.1. CRLs

En pocas palabras, las CRLs son estructuras de datos firmadas que contienen una lista de certificados revocados. La firma digital agregada en la CRL proporciona mecanismos de autenticidad e integridad. Siempre que las políticas lo permitan las CRLs pueden ser almacenadas en memoria y facilitar la verificación de certificados off-line. Normalmente el emisor del certificado y de la CRL son la misma autoridad, pero no siempre ocurre así.

Actualmente hay definidas 2 versiones de CRLs en el estándar X.509. La versión 1 presenta varios defectos: problemas de escalabilidad, posibilidad de ataques de sustitución que reemplazan una CRL por otra sin ser detectado,... La versión 2 corrige estos problemas mediante el mecanismo de las extensiones. En el caso de las CRLs hay dos tipos de extensiones: las aplicables sólo a una entrada de la lista y las aplicables globalmente a la CRL. Una extensión marcada como no crítica puede ser ignorada por la aplicación. Cuando una aplicación procesando una CRL no reconoce una extensión marcada como crítica que sólo aplica a un campo de la lista considerará el certificado como revocado y realizará las acciones adicionales que marque la política. Cuando una aplicación procesando una CRL no reconoce una extensión marcada como crítica que aplica globalmente a la CRL considerará los certificados identificados como revocados, pero además considerará que la lista no es completa y actuará adicionalmente como indique la política. Del mismo modo que ocurre con las extensiones de los certificados, las extensiones de CRL pueden ser adaptadas para un entorno concreto, tal es el caso de la recomendación RFC5280 del PKIX del IETF que ajusta las CRLs para su uso en Internet.

Estructura y semántica de las CRLs

A continuación se van a detallar los diferentes campos que constituyen las CRLs. La figura 1.2 representa el contenido de una CRL v2.

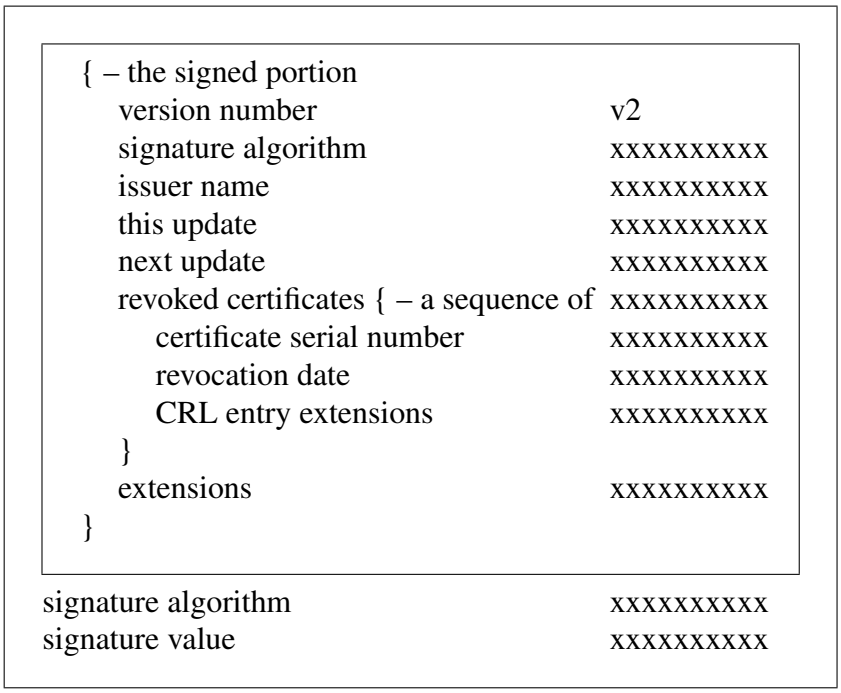

Figura 1.2: Elementos de una CRL X.509 v2

- *Version*. Indica la versión de la CRL. Si el campo no está presente indica que se trata de una CRL v1; si el campo está presente su valor debe ser el entero 1, indicando que se trata de una CRL v2.
- *Signature Algorithm*. Indica el OID del algoritmo usado para calcular la firma digital de la CRL. Debe coincidir con el campo Signature Algorithm perteneciente a la porción no firmada.
- *Issuer Name*. Se trata del DN del emisor de la CRL, es decir, quién firma la CRL. Debe estar siempre presente y ser único.
- *This Update*. Indica la fecha y hora en la que se emitió la CRL.

- *Next Update*. Campo opcional según X.509 que indica la fecha y hora en la que se emitirá la siguiente CRL.
- *Revoked Certificates*. Se trata de la lista de los certificados revocados. Cada entrada contiene el Serial Number del certificado revocado, la fecha y hora en la que se revocó el certificado y, opcionalmente, puede incluir extensiones aplicables a la entrada concreta de la lista.
- *Extensions*. Son las extensiones aplicables a la CRL globalmente.

El estándar X.509 define varias extensiones aplicables a una entrada de la lista. Estas extensiones permiten agregar información adicional a cada revocación y la más importantes son:

- *Reason Code*. Razón por la cual el certificado fue revocado (compromiso de la clave privada, compromiso de la CA, cambio de algún dato,...).
- *Certificate Issuer*. Es el nombre del emisor del certificado.
- *Hold Instruction*. Permite soportar la suspensión temporal de un certificado.
- *Invalidity Date*. Es la fecha y hora en la que el certificado deja de ser válido.

El estándar X.509 define varias extensiones aplicables globalmente a una CRL. A continuación se detallan algunas de ellas.

- *Authority Key Identifier*. Es el identificador único de la clave que debe usarse para verificar la firma digital. Distingue entre múltiples claves del mismo emisor de CRLs.
- *Issuer Alternative Name*. Contiene uno o más nombres alternativos del emisor de la CRL.
- *CRL Number*. Contiene un número de secuencia creciente para cada CRL emitida por un emisor de CRL. Permite detectar fácilmente cuando una CRL reemplaza a otra.
- *CRL Scope*. Proporciona un método flexible para particionar CRLs. Las CRLs se pueden partir de muchas maneras (tipo de certificado, razón de revocación, números de serie,...)

- *Status Referral*. Esta extensión está presente en CRLs que no llevan información sobre certificados revocados. Simplemente transporta información para asegurar que se usa la información de revocación apropiada. Por un lado proporciona información dinámica sobre la partición de CRLs y, por otro lado, publica una lista de CRLs actuales que se utilizan para establecer si ya se dispone de dicha información.

- *CRL Stream Identifier*. Identifica el contexto en el cual el CRL Number es único. Por ejemplo, un CRL Stream Identifier se puede asignar a cada CRL Distribution Point. Combinando el Stream Identifier con el CRL Number se proporciona un identificador único para cada CRL emitida por una CA.

- *Ordered List*. Indica si la lista de certificados revocados está en orden ascendente por Serial Number o fecha de revocación.

- *Delta Information*. Esta extensión proporciona la ubicación de la Delta CRL correspondiente a esta CRL.

- *Issuing Distribution Point*. Esta extensión identifica el CRL Distribution Point para esta CRL en concreto y el tipo de certificado contenido en la CRL (certificados de CA o certificados de entidad final).

- *Delta CRL Indicator*. Indica que esta CRL es una Delta CRL relativa a la CRL base referenciada.

- *Base Update*. Se usa en Delta CRLs para indicar la fecha y hora después de la cual la Delta CRL proporciona revocaciones.

- *Freshest CRL*. Identifica cómo obtener la Delta CRL más reciente para la CRL base que contiene esta extensión.

La mayoría de CAs emiten sus propias CRLs, es decir, la CA emite los certificados y las CRLs. Periódicamente la CA emite un única CRL que cubre toda su población de certificados. Tales CRLs se denominan *CRLs completas*. Sin embargo, el uso de las CRLs completas tiene importantes limitaciones: el tamaño de las CRLs puede crecer mucho, la periodicidad de publicación de las CRLs debe ser grande para evitar la degradación de los recursos de red. Esto obliga a buscar soluciones alternativas.

Para evitar que la CRL crezca demasiado, la CA puede dividir la población según el tipo de sujeto. Una CARL es una CRL dedicada exclusivamente a revocar certificados de CAs. En una PKI jerárquica la revocación de una CA superior impacta en todas las CAs subordinadas y en todas las entidades finales que caen bajo cualquiera de las CAs afectadas. Una EPRL es una CRL dedicada exclusivamente a la revocación de entidades finales. Los *CRL Distribution Points* también permiten que la información de revocación de una única CA se distribuya en varias CRLs. Los certificados apuntan a la ubicación de la CRL, de modo que la aplicación que valida el certificado no necesita conocer a priori dónde se encuentra la CRL. Un inconveniente asociado con el uso de los CRL Distribution Points es que una vez se ha emitido el certificado el CRL Distribution Point está fijado durante toda la vida del certificado. Las extensiones *CRL Scope* y *Status Referral* permiten flexibilizar el mecanismo de los CRL Distribution Points.

El estándar X.509 define dos tipos adicionales de CRLs: las *Delta CRL* y las *CRLs indirectas*. Una Delta CRL sólo contiene información de revocación no disponible cuando se emitió una CRL base o bien desde un determinado instante de tiempo. Esto permite la publicación de Delta CRLs relativamente pequeñas que pueden ser emitidas con mucha mayor frecuencia que la CRL base optimizando los objetivos, a menudo contradictorios, de rendimiento e información actualizada. Para disponer de toda la información de revocación disponible en un instante de tiempo dado basta con la CRL base y la última Delta CRL. No hay que acumular las Delta CRLs emitidas con anterioridad. El coste adicional de las Delta CRLs es que la validación de certificados es más compleja. Por otro lado, las CRL indirectas permiten que información de revocación de múltiples CAs se emita en una única CRL. Las CRL indirectas reducen el número de CRLs que se necesitan recuperar cuando se está validando un certificado. Por ejemplo, un único dominio PKI puede tener varias CAs y en lugar de usar varias CRLs, una para cada CA, el dominio puede usar una CRL Indirecta para mejorar la eficiencia. El estándar X.509 también introduce el concepto de *Delta CRL indirecta* que combina los dos conceptos anteriores.

1.4.2. Mecanismos de consulta on-line

La principal diferencia entre los mecanismos de publicación periódica y los mecanismos de consulta on-line consiste en que la parte que analiza un

certificado debe estar on-line. Los mecanismos de publicación periódica son más adecuados para operar off-line porque la información de revocación puede guardarse en memoria. El grupo PKIX del IETF ha dedicado un esfuerzo significativo a definir el OCSP (On-line Certificate Status Protocol, RFC2560) que constituye el mecanismo de consulta on-line más usado. Recientemente el grupo PKIX ha explorado los requisitos de *Delegated Path Discovery* (DPD) y *Delegated Path Validation* (DPV) y ha desarrollado el protocolo *Simple Certificate Validation Protocol* (SCVP).

OCSP

OCSP es un protocolo solicitud-respuesta relativamente simple que permite una manera de obtener información de revocación on-line de una entidad fiable denominada OCSP responder. Una solicitud OCSP consiste en la versión del protocolo, el tipo de servicio solicitado y uno o más identificadores de certificado. El identificador de certificado consiste, a su vez, en el hash del DN del emisor, el hash de la clave pública del certificado y el Serial Number del certificado. Las respuestas consisten en el identificador del certificado, el estado del certificado ('good', 'revoked' o 'unknown') y el intervalo de validez de la respuesta. Las respuestas OCSP deben estar firmadas digitalmente para que la parte solicitante pueda confiar en ellas. Según el protocolo las solicitudes pueden estar firmadas o no.

OCSP sólo verifica si el certificado ha sido revocado o no. No verifica el periodo de validez ni si el certificado es válido en el contexto de uso. Por otro lado, OCSP no es más que un protocolo y no especifica la infraestructura que respalda el servicio; es decir, no elimina necesariamente las CRLs u otros métodos para recoger la información de revocación de los certificados.

SCVP

SCVP permite DPV y DPD en el entorno de Internet. DPV permite delegar en otro componente el proceso de validación de un certificado. Por otro lado, DPD permite delegar el proceso de la construcción del camino de los certificados. Esto puede ser particularmente útil en el caso de dispositivos en los que la validación local es inviable por motivos de rendimiento como es el caso de los teléfonos móviles.

1.5. Componentes de una PKI

En esta sección se introducen los componentes de una PKI y se describen sus funcionalidades. Las funcionalidades deben estar presentes en cualquier PKI, pero las implementaciones específicas pueden agruparlas de forma diferente. Por ejemplo, la Certification Authority (Autoridad de Certificación, CA) y la Registration Authority (Autoridad de Registro, RA) en ocasiones se combinan en un único componente. Esto no afecta *qué* funciones se realizan, sino *dónde* se realizan. Las PKIs se construyen con varios componentes, cada uno diseñado para realizar unas pocas tareas particularmente bien.

La figura 1.3 presenta un modelo simplificado de los componentes de una PKI basado en el que se describe en el RFC5280 del PKIX del IETF:

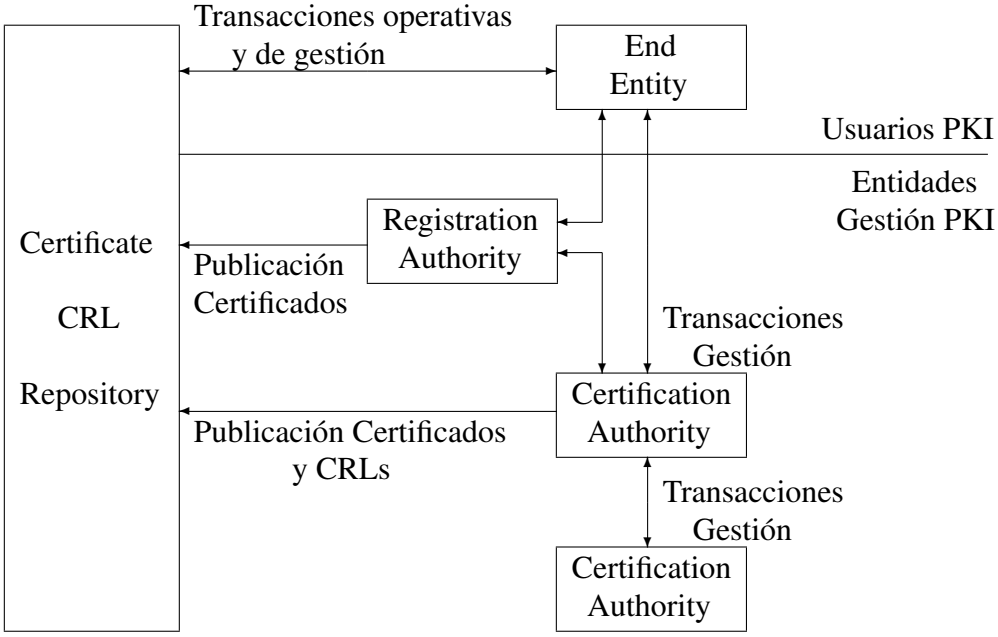

Figura 1.3: Esquema simplificado PKI

A continuación se explica el esquema someramente. Los usuarios finales (end entities), usando transacciones de gestión, envían su petición de certificado a la RA para su aprobación. Si la petición es aprobada, se pasa a la CA para ser firmada. La CA revisa la petición de certificado y, si supera la

revisión, se firma la petición y se genera el certificado. Para publicar el certificado, la CA lo envía al repositorio de certificados. El diagrama también muestra que los usuarios finales pueden comunicarse directamente con la CA de manera que toda la funcionalidad está implementada en la CA. Del mismo modo, el diagrama muestra que CA y RA envían los certificados al repositorio. La implementación debe escoger una de las alternativas.

La revocación de certificados sigue un curso similar al de la generación. Los usuarios finales piden a la RA que revoque su certificado, la RA decide y reenvía la solicitud a la CA. La CA actualiza la Certificate Revocation List (Lista de Certificados Revocados, CRL) y la publica en el repositorio de la CRL.

Finalmente, los usuarios finales pueden verificar la validez de un certificado específico usando un protocolo operativo.

1.5.1. Certification Authority

La CA es el componente esencial de una PKI. Una CA es un conjunto de hardware, software y el personal que los opera. Una CA realiza cuatro funciones básicas:

- *Emisión de certificados*, crea los certificados y los firma.
- *Mantener información del estado de los certificados y emitir CRLs*.
- *Publicar los certificados y CRLs* de manera que los usuarios puedan obtener la información que necesitan para implementar los servicios de seguridad.
- *Mantener archivos sobre la información de estado de los certificados expirados o revocados que emitió*.

Emisión de certificados

Una CA puede emitir certificados a usuarios, otras CAs o ambos. Cuando una CA emite un certificado está afirmando que el sujeto (entidad nombrada en el certificado) tiene la clave privada que corresponde a la clave pública del certificado. Si la CA incluye información adicional en el certificado, la CA está dando por cierto que dicha información corresponde al sujeto. Esta

información adicional podría ser información de contacto (por ejemplo, e-mail) o información de política (por ejemplo, los tipos de aplicaciones en los que se puede usar la clave pública). Cuando el sujeto del certificado es otra CA, la CA emisora está afirmando que se puede confiar en los certificados emitidos por la CA sujeto.

La primera responsabilidad de una CA es proteger su clave privada. Si un atacante consiguiera la clave privada de una CA podría suplantarla y emitir certificados como si fuera la misma CA. Para proteger la clave privada la CA utiliza un módulo criptográfico. Los módulos criptográficos generan claves, implementan algoritmos criptográficos y protegen las claves. Pueden implementarse en hardware, software o una combinación de ambos. Los módulos criptográficos de hardware (smart cards, tarjetas PCMCIA, ...) realizan las operaciones criptográficas en un procesador externo y guardan la clave privada fuera de la memoria del sistema. El *National Institute of Standards and Technology* (NIST) desarrolló el estándar FIPS 140, *Security Requirements for Cryptographic Modules*, que especifica cuatro niveles de seguridad progresivos para los módulos criptográficos. Se recomienda que una CA utilice siempre un módulo criptográfico de hardware de, al menos, nivel 2 de acuerdo con el FIPS 140 para generar y proteger su clave privada.

La segunda responsabilidad de una CA es verificar que la información del certificado (información del sujeto, información del contacto, información de política, ...) es cierta. Si bien la CA puede verificar mediante la firma digital que el sujeto posee la clave privada correspondiente, el resto de la información debe verificarse externamente ya que depende de información proporcionada por agentes externos a la CA.

La tercera responsabilidad de una CA es asegurar que todos los certificados y CRLs que emite cumplen con un perfil. Por ejemplo, si una CA se diseña para emitir certificados para e-mail, no puede emitir un certificado que autorice al sujeto a la firma de contratos. Para asegurar el cumplimiento de esta responsabilidad, la CA debe verificar que todos y cada uno de los certificados y CRLs que firma cumplen con el perfil. Por otro lado, la CA también debe proteger la integridad del perfil y restringir su acceso. La restricción del acceso puede ser física (acceso a través de tarjeta al recinto), lógica (firewall) o procedimental (se requieren dos miembros del personal que atiende la CA para modificar el sistema).

Mantenimiento de la información de estado y emisión CRLs

Al igual que con los certificados, los contenidos de las CRLs deben ser correctos para ser útiles. Una omisión en una CRL puede provocar que un usuario acepte un certificado revocado, mientras que una revocación incorrecta puede resultar en una denegación de servicio. La cuarta responsabilidad de una CA es mantener una lista de certificados en los que no se debe confiar. Proteger esta información es similar a proteger el perfil, mientras que modificar el estado de un certificado depende de información proporcionada desde el exterior de la CA.

Publicación de certificados y CRLs

Los certificados y CRLs que emite una CA sólo son útiles si están disponibles de manera que los usuarios puedan implementar los servicios de seguridad que requieren. La quinta responsabilidad de una CA es distribuir sus certificados y CRLs. En general, la distribución de los certificados es más un problema de rendimiento y disponibilidad que de seguridad; no se requiere restringir el acceso a certificados y CRLs puesto que tal información no es secreta. Sin embargo, en ocasiones, la CA puede querer denegar el acceso a los certificados a personas ajenas a la organización por simples motivos de seguridad.

Mantenimiento de archivos

Finalmente, la CA necesita mantener información para identificar la firma de un documento antiguo cuyo certificado ya ha expirado. Para soportar este objetivo el archivo debe demostrar que el certificado era válido en el momento en que se firmó del documento. Esto puede requerir el uso de un servidor de tiempo criptográfico. La sexta responsabilidad de una CA es el mantenimiento de suficiente información de archivo para establecer la validez de certificados después de que hayan expirado. La principal dificultad de esta función es que la información debe mantenerse durante largos periodos de tiempo.

Delegación de responsabilidad

Es difícil diseñar un sistema que satisfaga simultáneamente todos los requisitos. Una CA suele cumplir los más prioritarios y delegar el resto. La responsabilidad principal de una CA es proteger la clave o claves privadas

usadas para firmar certificados y CRLs. Para satisfacer este requisito la CA debe construir un perímetro con controles de seguridad física, tecnológica y procedimental. Este perímetro permite también alcanzar los requisitos tercero y cuarto.

Este perímetro de seguridad impide el cumplimiento de las restantes responsabilidades. Los tres componentes restantes de la infraestructura se diseñan para aceptar esas responsabilidades en lugar de la CA. Una entidad que verifica el contenido de los certificados se denomina Registration Authority (RA). Una RA también puede asumir algunas de las responsabilidades para revocar certificados. En la práctica, una CA puede disponer de muchas RAs; por ejemplo, una para cada grupo de usuarios. Una entidad que distribuye certificados y CRLs se llama repositorio y se diseña para maximizar rendimiento y disponibilidad. Los repositorios a menudo se duplican para maximizar la disponibilidad e incrementar el rendimiento. La entidad que proporciona almacenamiento seguro a largo plazo se denomina archivo. Esta entidad no requiere un elevado rendimiento ni tampoco que se mantengan múltiples archivos.

1.5.2. Registration Authority

El propósito de una RA es verificar el contenido de los certificados en lugar de la CA. De igual modo que una CA, una RA es una colección de hardware, software y las personas que lo operan. Cada CA mantiene una lista de RAs acreditadas y verificando la firma de una RA en un mensaje una CA puede estar segura de que una RA acreditada proporcionó la información y, por tanto, es fiable. Por consiguiente, es importante que una RA proporcione protección adecuada para su clave privada. Se recomienda que las RAs usen módulos criptográficos de hardware validados según el estándar FIPS 140.

Hay dos modelos básicos para que una RA verifique el contenido de un certificado. En el primer modelo la RA recoge y verifica la información antes de presentar a la CA la solicitud para el certificado. En el segundo modelo, la CA recibe una solicitud de certificado que envía a la RA. La RA revisa el contenido y determina si la información es correcta. La RA responde a la petición de la CA con un simple 'Sí' o 'No'.

El primer modelo se usa cuando el usuario se persona en la RA. En tal caso, se puede verificar la identidad del usuario mediante el DNI o el permiso de conducir. Si el usuario ha generado su par de claves, la RA solicita el

certificado a la CA con la información apropiada. Alternativamente, la RA puede proporcionar un valor secreto al usuario. El usuario genera su par de claves, emite la solicitud a la CA y se autentifica con el valor secreto. Este modelo puede también usarse cuando la RA posee el módulo criptográfico de hardware correspondiente al usuario. En general es preferible que los usuarios se generen su propio par de claves, pero la generación de claves puede ser una tarea muy exigente en términos de recursos computacionales. Para limitar el coste del módulo criptográfico puede ser deseable generar pares de claves usando un hardware especial y cargar la clave privada en el módulo de hardware del usuario en la RA. En este caso es la RA quien genera el par de claves y crea una solicitud de certificado con la clave pública y el resto de información. Tras obtener el certificado de la CA, la RA entrega el módulo criptográfico al usuario.

El segundo modelo se usa cuando el usuario no puede ser identificado de antemano y genera la solicitud de certificado. En este caso el usuario solicita que el certificado contenga cierta información. La CA puede determinar que el usuario posee las claves pública y privada, pero no puede saber si el resto de la información es correcta. La CA traspasa la solicitud a la RA que la aceptará o rechazará y proporcionará el resultado a la CA.

1.5.3. Repositorio

Un repositorio acepta certificados y CRLs de una o más CAs y los hace disponibles a las partes que necesitan implementar servicios de seguridad bajo petición. Los repositorios se diseñan para proporcionar máxima disponibilidad y rendimiento, ya que los mismos datos establecen su integridad. Los repositorios necesitan restringir el conjunto de usuarios que pueden actualizar la información, ya que, de lo contrario, un atacante podría sustituir los certificados con basura y provocar un ataque de denegación de servicio.

1.5.4. Archivo

Un archivo asume la responsabilidad del almacenamiento a largo plazo de información en lugar de la propia CA. Un archivo declara que la información era correcta en el momento en que se recibió y que no ha sido modificada. El archivo protege la información a través de mecanismos técnicos y procedimientos. Si se suscita una disputa, la información del archivo puede usarse

para verificar firmas de documentos viejos en fechas posteriores.

1.5.5. Usuarios

Hay dos tipos de usuarios soportados por una PKI. Los poseedores del certificado son los sujetos del certificado y mantienen la clave privada. Las partes confiantes usan la clave pública de un certificado para verificar la firma o cifrar datos. En la práctica, la mayoría de entidades soportan ambos papeles. Del mismo modo, CAs y RAs son también usuarios ya que generan y verifican firmas y transmiten claves entre ellas mismas o con los propios usuarios.

Poseedores de certificados

Los poseedores de certificados obtienen certificados de la infraestructura y usan sus claves privadas para implementar servicios de seguridad. Generan firmas digitales, descifran datos (por ejemplo claves simétricas) y usan sus claves privadas para establecer claves simétricas a través de protocolos de acuerdo de claves. Para cumplir estos objetivos el poseedor de un certificado debe realizar las siguientes acciones:

- Identificar la CA que emite los certificados.
- Solicitar el certificado directamente o a través de una RA.
- Incluir el certificado en las transacciones que así lo requieran.

En ocasiones los poseedores de un certificado necesitarán interactuar con el repositorio para obtener su certificado, aunque no de una manera regular.

Partes confiantes

Las partes confiantes usan la PKI para implementar servicios de seguridad utilizando la clave pública en el certificado. Pueden verificar firmas digitales, cifrar datos (claves simétricas) y usar la clave pública para establecer claves simétricas a través de protocolos de acuerdo de claves. Para implementar estos servicios de seguridad, una parte confiante debe realizar las siguientes acciones:

- Identificar una CA como su punto inicial de confianza.

- Verificar firmas de certificados y CRLs.
- Obtener certificados y CRLs del repositorio.
- Construir y validar caminos de certificación.

Una parte confiante interactúa con el repositorio regularmente. Sus interacciones con las CAs se limitan a la selección de los puntos de confianza y no interaccionan con las RAs.

Capítulo 2

Evaluación

2.1. Introducción

En última instancia, el despliegue de una PKI en una organización está motivado por necesidades de seguridad avanzada en las comunicaciones a través de la red o en el almacenamiento de información.

Durante la fase de evaluación de la necesidad de una PKI se analizan los requisitos de seguridad presentes y futuros de la organización. Previamente puede ser necesario recoger información mediante una auditoría de seguridad o un test de penetración.

A continuación nos centraremos en las áreas claves de la fase de evaluación: análisis de los requisitos de negocio, beneficios que aporta una PKI, documentos CP y CPS y estudio de costes. También se proporcionarán criterios sobre la externalización y sobre la selección del producto o el proveedor del servicio.

2.2. Análisis de los requisitos de negocio

¿Cómo afecta la organización o empresa a la PKI? La organización o empresa va a afectar de distintos modos a la PKI, pero la influencia más decisiva es mediante las razones de negocio que provocarán su despliegue. La razón o razones de negocio que motivan el despliegue de una PKI tendrán impacto en cada etapa del proyecto y determinarán la inversión que una empresa está dispuesta a realizar ya que ésta dependerá de la importancia de los procesos cuyo nivel de seguridad se quiere aumentar por medio de la PKI.

Hay muchos artículos en Internet que dan razones muy variadas que justifican la implantación de una PKI. Por ejemplo, el Dartmouth College promueve el uso de PKIs en instituciones educativas superiores para mejorar la seguridad y fomentar una colaboración mucho más estrecha entre ellas que la actual. Las redes educativas son muy abiertas y están expuestas a hackers, virus y spam; además, los usuarios suelen tener comportamientos arriesgados como, por ejemplo, compartir passwords. Una PKI permite combatir estos inconvenientes. El gigante industrial Johnson & Johnson decidió implantar una PKI para reducir los gastos derivados del mantenimiento de las passwords. Según sus estimaciones costaba 37$ por empleado y año soportar los cambios y resets de passwords. El cambio fue muy complejo ya que se incluyó adaptar aplicaciones de software de diferentes fabricantes (Siebel, SAP, J.D. Edwards,...) e implantar un servicio de directorio (Active Directory de Windows 2000). Otra causa común de implantación de PKIs, sobretodo en organizaciones gubernamentales, es la firma digital. La posibilidad de firmar digitalmente un documento electrónico permite ahorrar tiempo y papel. El Bank of Bermuda tenía como objetivo proporcionar acceso electrónico seguro y económico a sus servicios tanto a personal propio como a clientes en cualquier lugar del mundo y en cualquier momento.

En ocasiones, los requisitos de seguridad de algunas empresas u organizaciones no precisan una solución basada en certificados y se pueden resolver de forma más simple. Por otro lado, no todas las soluciones basadas en certificados requieren el despliegue de una PKI, algunas compañías pueden decidir simplemente comprar un conjunto limitado de certificados de una Autoridad de Certificación (CA) comercial.

La organización tiene otras implicaciones en los requisitos que debe cumplir la PKI:

- *Disponibilidad*. ¿Qué disponibilidad necesita la PKI en la organización? Pensemos en un banco, por ejemplo, que requiere que sus servicios funcionen 24 horas al día los 365 días del año. Esto repercutirá en el diseño de las CAs, directorio,...

- *Escalabilidad*. ¿Necesita la PKI ser escalable? ¿Crecerá rápidamente el número de certificados? ¿Se prevén fusiones con otras empresas? ¿Se desplegarán muchas aplicaciones PKI en el futuro?

- *Rendimiento*. Las operaciones criptográficas pueden requerir hardware especial o obligar a actualizar el hardware.

- *Coste*. Las organizaciones o empresas tienen un presupuesto limitado que puede condicionar la elección de la PKI.

Todas estas implicaciones hacen que la implantación de una PKI en una organización sea un proceso fuertemente depediente de la organización y prácticamente único.

Los beneficios que aporta una PKI se hallan íntimamente relacionados con el análisis de requisitos.

2.3. Beneficios de una PKI

En este apartado se pretende aportar las razones que justifican la implantación de una PKI, tanto las razones técnicas como las puramente de negocio. Se pretende responder a la pregunta siguiente: ¿cómo mejorará una organización la implantación de una PKI? Una implantación con el único propósito de disponer de la última tecnología está condenada al fracaso. El diseñador de una solución PKI debe tener una visión clara de cómo la PKI ayuda a mejorar la organización o empresa. Esta visión puede focalizar la implantación en un área que proporcione resultados tangibles inmediatos y que asegure el éxito del proyecto. Muchas organizaciones, particularmente empresas, no se conformarán con simples razones técnicas para llevar a cabo la implantación. Las empresas desean conocer razones de negocio, es decir, ¿cómo ahorrará dinero una PKI, o más concretamente, una CA o una smart card? Tal vez, la etapa más crítica en cualquier proyecto es proporcionar sólidas justificaciones de negocio para llevarlo a cabo.

Los beneficios que una PKI ofrece se derivan del propio concepto de PKI y de los servicios que ofrece. Una PKI es una solución global de seguridad, no un conjunto de soluciones puntuales diferentes, y ofrece una única infraestructura de seguridad que puede ser usada por muchas aplicaciones en los entornos más heterogéneos. Específicamente, ofrece servicios de confidencialidad, integridad, autenticación y no repudio en numerosos contextos.

He aquí una lista de los beneficios que ofrece una PKI:

- *Gestión de passwords y Single-Sign-On*. Hay muchos problemas ocasionados por el sistema tradicional de usernames y passwords. Una PKI resuelve estos problemas de una manera consistente y sencilla para los usuarios y administradores.

- *Firmas digitales*. Una PKI permite firmar digitalmente documentos. Las firmas tienen un reconocimiento legal. En conjunto se consigue sustituir el papel con formularios electrónicos, más velocidad y trazabilidad en los procesos de negocios y una seguridad mejorada en las transacciones electrónicas.

- *Cifrado*. Fácil cifrado de datos para cada individuo (sin intercambio previo de información) mediante el acceso al certificado que contiene la clave pública.

- *Comodidad del usuario*. Menos passwords. Mecanismo consistente de autenticación que basta con aprender una vez. Procedimientos sencillos para cifrar, firmar y autenticar.

- *Administración coherente de seguridad en la empresa*. Emisión y revocación centralizada de credenciales de usuario. Identificación consistente de usuario cuando se emiten las credenciales. Idéntico mecanismo de autenticación para todas las aplicaciones o servicios de red. Aprovechamiento de la inversión en smart cards o tokens USB al ser usados por muchas aplicaciones.

- *Interoperabilidad con otras instituciones*. La confianza entre organizaciones y/o empresas permite firmar y cifrar correos, firmar documentos, autenticación en aplicaciones compartidas,...

- *Solución basada en estándares*. Los estándares proporcionan interoperabilidad entre fabricantes diferentes. Muchas implementaciones disponibles. Los estándares permiten que haya código libre.

- *Amplio soporte*. En sistemas operativos: Windows, Linux, UNIX,.... Almacenamiento de claves en software y hardware. En aplicaciones: Apache, IIS, Oracle, SSL,.... Código abierto y comercial.

- *Entornos variados*. Las PKI son soluciones ampliamente aceptadas en la industria (Johnson & Johnson, Microsoft,...), organizaciones gubernamentales,...

Estas ventajas técnicas se traducen en ventajas de negocio, aunque su cuantificación concreta es difícil:

- *Mejoras en la eficiciencia del workflow*. Se pueden conseguir ahorros significativos de tiempo mediante el manejo electrónico de documentos.

- *Optimización de los recursos humanos*. El usuario puede centrarse en su trabajo en lugar de gastar tiempo en detalles asociados con la infraestructura de seguridad.

- *Reducción de los recursos humanos*. La operación de una arquitectura unificada en lugar de múltiples soluciones puntuales requiere menos recursos administrativos.

- *Reducción del papel*. Se puede ahorrar en costes de material, en el espacio necesario para almacenarlo, en reducción de residuos y en menor intrusión ambiental.

- *Menos carga administrativa*. Los usuarios finales requieren menos asistencia (help-desk,...).

- *Reducción de pérdidas por robo electrónico*. Los datos corporativos están protegidos, lo que reduce el riesgo de que sean revelados sin autorización.

- *Ahorros en telecomunicaciones*. La capacidad de crear redes privadas virtuales (Virtual Private Networks, VPN) sobre una red pública como Internet es más barata que alquilar líneas privadas.

- *Generación de ingresos*. Una PKI puede usarse para generar ingresos. Por ejemplo, una organización financiera puede ofrecer servicios de validación de transacciones basados en firmas digitales y certificados.

En cualquier caso, el robo y fraude electrónico van en aumento y deben buscarse soluciones. Las empresas perciben la seguridad como algo necesario y, por tanto, le dedican cierta atención. Cualquier solución global de seguridad debe contemplar los recursos corporativos y las comunicaciones, tanto externas como internas. ¿Qué ocurriría si le robasen el portátil al CEO de la empresa y la información sensible no estuviera cifrada? Es difícil de cuantificar el daño económico que sufriría la empresa si esa información cayera en malas manos. Se puede argumentar que no hace falta una PKI para cifrar la información de un portátil, pero una PKI ofrece también la posibilidad de

recuperar la información cifrada si el CEO, por ejemplo, sufriera una incapacidad transitoria. El valor de una PKI que protege y permite recuperar información crítica es muy elevado, incluso si sus beneficios cuantitativos son casi imposibles de medir con precisión.

Una PKI es una infraestructura y, como tal, sólo produce beneficios cuando es utilizada. De hecho, son las aplicaciones quienes se conectan a la infraestructura, la utilizan y provocan beneficios. Los tipos de certificados que serán necesarios y las entidades a las cuales se emitirán (usuarios, máquinas,...) dependen de las aplicaciones que vaya a soportar la PKI. Es importante conocer las principales aplicaciones que pueden aprovechar la PKI y proporcionar una seguridad fuerte ya que la organización o empresa querrá usar algunas de ellas. Algunas aplicaciones están inmersas en el sistema operativo y otras no. He aquí algunas de las principales aplicaciones:

- *Web segura*. En una intranet corporativa o en una extranet se pueden usar certificados para proporcionar seguridad fuerte mediante los protocolos SSL y TLS. Ambos protocolos proporcionan autenticación de cliente, autenticación de servidor y confidencialidad de datos.

- *Correo seguro*. El protocolo Secure/Multipurpose Internet Mail Extensions (S/MIME) también está basado en criptografía de clave pública y certificados. Este protocolo permite firmar y cifrar mensajes. Muchas aplicaciones de e-mail proporcionan un sistema dual de claves para firmar y cifrar.

- *Cifrado del sistema de ficheros*. Proporciona cifrado a nivel del sistema de ficheros. Como medida de seguridad, conviene que permita la recuperación de los datos por una persona adicional.

- *Firma de código*. Protege contra descargas de código alterado (hackers) de websites.

- *Smart card logon*. Proporciona autenticación fuerte de 2 factores (posesión de la smart card y conocimiento del PIN (Personal Identification Number)). A diferencia de las passwords, los PINs no se envían a través de la red, lo cual es una medida adicional de seguridad.

- *Virtual Private Network*. IPSec es un protocolo que funciona a nivel IP con certificados y se usa para autenticar los extremos de la comunicación.

- *Cualquier aplicación*. Los fabricantes proporcionan APIs con las que adaptar cualquier aplicación para que use la PKI.

2.4. CP y CPS

¿Qué grado de confianza otorgo a un certificado concreto?

Hay dos tipos de documentos que describen las políticas y procedimientos asociados a una PKI. El primer documento se denomina *Certificate Policy* (Política de Certificados, CP) y el segundo *Certification Practice Statements* (Prácticas de Certificación, CPS). Estos documentos tienen un formato común pero diferentes objetivos y audiencias. Existe una relación directa entre el contenido de las extensiones de políticas de los certificados X.509 v3 y los documentos CP.

Una política es un conjunto de reglas establecidas para gobernar un cierto aspecto del comportamiento de una organización, concretamente describen *qué* debe hacerse para alcanzar, por ejemplo, los objetivos de la organización. Una *Security Policy* (Política de Seguridad, SP) describe objetivos, responsabilidades y requisitos generales para la protección de recursos específicos (por ejemplo, algunos datos). Las políticas de seguridad se implementan a través de una combinación de mecanismos y procedimientos. Los mecanismos y procedimientos describen *cómo* se alcanzan los requisitos de seguridad descritos en la política.

Una CP es un documento de alto nivel que describe una política de seguridad para emitir certificados y mantenerlos. Describe las operaciones de CAs y sus componentes de soporte y las responsabilidades de los usuarios al solicitar y usar los certificados. Una CP puede describir el comportamiento de una única CA y sus componentes de soporte, este es el caso cuando una única CA da servicio a una organización. Puede darse también el caso de que una única CP describa el comportamiento de múltiples CAs y sus componentes de soporte. Es el caso de múltiples CAs mantenidas por una organización a través de una PKI en malla. También una CP puede describir una o varias PKI jerárquicas. Una CP tiene distintos usos. En primer lugar, se usa como guía para el desarrollo de los CPSs de cada CA. Otras organizaciones revisarán la CP antes de hacer una cross-certificación. Los auditores de seguridad tomarán la CP como base de las auditorías. Los propietarios de una aplicación estudiarán la CP para determinar si los certificados son apropiados para la aplicación.

Un CPS es un documento con un alto grado de detalle que explica cómo una CA implementa una CP específica. Un CPS determina los procedimientos con los que se usarán productos específicos. Un CPS incluye suficientes detalles operativos para demostrar que una CP se satisface con una combinación de mecanismos y procedimientos. Un CPS se aplica a una única CA y sus componentes de soporte y puede considerarse como el manual de operaciones de la CA. Los auditores de seguridad usan los CPS como un complemento a las CP durante las auditorías de seguridad. Sin embargo, un CPS no necesita ser publicado. Cada CP puede ser implementada por muchos diferentes CPSs y un CPS puede cumplir los requisitos de más de una CP.

2.4.1. Formato y contenido de CP y CPS

El documento RFC3647 (antiguo RFC2527) *Internet X.509 Public Key Infrastructure Certificate Policy and Certification Practices Framework* establece el formato recomendado para CPs y CPSs. La mayoría de CPs y CPSs se escriben siguiendo el esquema del estándar porque tiene ventajas. Al adherirse a un formato bien definido es difícil olvidar algo importante. Sin embargo, el estándar es flexible y permite añadir al escritor de estos documentos puntos adicionales para cumplir los requisitos particulares de su organización. Los CPs de distintas organizaciones, al seguir un formato común, se pueden comparar fácilmente y así evaluar sus diferencias. Los documentos se usan para establecer mapeos de políticas que pueden representarse en una extensión de los certificados. Puesto que CPs y CPSs siguen el mismo formato es fácil revisarlos en conjunto y asegurarse de que los mecanismos y procedimientos especificados en el CPS implementan fielmente la CP.

El RFC3647 propone dividir en nueve apartados los CPs y CPSs:

- Introducción
- Publicación y repositorio
- Identificación y autenticación
- Requisitos operativos del ciclo de vida de un certificado
- Controles físicos, operativos y de personal
- Controles de seguridad técnicos

- Perfiles de certificados, CRLs y OCSP
- Auditoría de conformidad
- Otros asuntos de negocio y legales

Introducción

Este componente proporciona una introducción general a la PKI. Puede contener los diferentes niveles de seguridad de la PKI, un diagrama de la propia PKI, descripciones de los participantes en la PKI y los papeles que realizan, listar los tipos de aplicaciones para las cuales son adecuados los certificados emitidos y listar los tipos de aplicaciones para las cuales está prohibido el uso de los certificados. También proporciona información de contacto de los autores del documento.

Publicación y repositorio

Este componente identifica las entidades que operan los repositorios en la PKI, la responsabilidad relativa a la publicación de información de cada participante en la PKI, cuándo y con qué frecuencia debe publicarse la información y, finalmente, el control de acceso a la información publicada (CPs, CPSs, certificados, CRLs,...).

Identificación y autenticación

Esta sección describe los procedimientos usados para autenticar la identidad u otros atributos de un usuario final solicitante de un certificado a la CA o RA previo a la emisión del certificado o a la reemisión. También se ocupa de la verificación de identidades para revocar certificados. Además, explica los procedimientos para autenticar la identidad y los criterios para aceptar solicitantes que quieran ser CAs, RAs u otras entidades operando o interoperando con la PKI. Este componente también se ocupa de las convenciones que deben seguir los nombres en los certificados (formatos de nombres usados y su interpretación, si deben ser significativos o no,...).

Requisitos operativos del ciclo de vida de un certificado

Este apartado especifica requisitos impuestos a los distintos participantes de una PKI. Describe, por ejemplo, el proceso usado por un sujeto para pre-

sentar una solicitud de certificado y sus responsabilidades relacionadas con el proceso. También explica el procedimiento para procesar una solicitud de certificado, las acciones realizadas por una CA durante la emisión del certificado, mecanismos de notificación, aceptación y publicación del certificado, responsabilidades de los poseedores y usuarios de los certificados relativas al uso del propio certificado y la clave privada, procedimientos relacionados con la renovación, modificación, revocación y suspensión de certificados así como la generación de nuevos pares de claves. Además esta sección trata de los servicios disponibles para verificar el estado de un certificado y procedimientos de custodia y recuperación de claves.

Controles físicos, operativos y de personal

Esta sección describe controles de seguridad no técnicos (físicos, procedimentales y de personal) usados por la PKI para realizar las funciones de generación de claves, autenticación de identidades, emisión y revocación de certificados, auditorías y archivo. Los controles físicos se refieren a las facilidades para albergar los diferentes equipos de la PKI como habitaciones con control de acceso, instalaciones para prevenir incendios, almacenamientos de backups,.... Los controles operativos identifican los diferentes roles (administrador, auditor, operador,...) y sus responsabilidades. Los controles de personal se refieren a las cualificaciones necesarias para efectuar cada rol, formación interna que debe seguir cada rol, sanciones por acciones no autorizadas, controles del personal subcontratado. Esta sección también define los eventos que se deben registrar, los sistemas de auditoría y las políticas de archivo de los eventos. Finalmente, esta sección se ocupa de los procedimientos de notificación y recuperación en el caso de un compromiso o desastre.

Controles de seguridad técnicos

Este apartado define las medidas de seguridad que adoptan las CAs y otros participantes de la PKI relativos a la protección de claves. También impone restricciones sobre los repositorios, sistemas operativos de los diversos componentes de la PKI y entornos de desarrollo y mantenimiento de software. Otro aspecto crucial de los controles de seguridad técnicos es la seguridad de red. En general conviene aislar las CAs con un firewall.

Perfiles de certificados, CRLs y OCSP

Esta sección especifica el formato de los certificados y CRLs. También debe especificar si se usa OCSP. Esto incluye información sobre perfiles, versiones y extensiones. Un aspecto importante de este apartado es el conjunto de algoritmos que se usarán para identificar las claves públicas y las firmas. Esta sección debe revisarse cuidadosamente si se pretende interconectar dos PKIs.

Auditoría de conformidad

Este apartado cubre los temas relativos a auditorías y evaluaciones de la PKI: frecuencia, identidad y cualificaciones del personal que realiza las auditorías o evaluaciones, acciones a tomar como consecuencia de deficiencias detectadas, personas autorizadas a revisar las auditorías,....

Otros asuntos de negocio y legales

Esta sección incluye las tasas de los diferentes servicios, la responsabilidad financiera de los participantes en el mantenimiento de los recursos y la jurisdicción legal en caso de disputas. Otros aspectos considerados en esta sección son: confidencialidad de la información, privacidad de la información personal, derechos de propiedad intelectual, garantías, limitaciones de responsabilidad, indemnizaciones,....

2.5. Estudio de costes

La mayoría de estudios coinciden en que una PKI debe verse como una inversión, ya que provocará un retorno financiero, y no como un simple gasto. El objetivo de esta sección no es proporcionar una fórmula general o un conjunto de fórmulas para determinar la inversión (TCO, Total Cost of Ownership) que una entidad debe realizar en PKI, ni tampoco para determinar el retorno de la inversión (ROI, Return On Investment) ya que no existen. Esto es debido al amplio abanico de posibilidades que ofrece una PKI y a las variadas necesidades que tienen las organizaciones o empresas. Además, una PKI es una infraestructura y, en general, no es fácil calcular el ROI de una infraestructura. Al mismo tiempo cabe plantearse otras preguntas más profundas,

¿se plantea una entidad cuál es el ROI cuando compra teléfonos, faxes o instala una red de PCs? La respuesta es no, simplemente es una inversión que se considera necesaria. Siguiendo esta argumentación se concluiría que no tiene sentido calcular el ROI de una PKI. El ROI de una PKI lo provocan las aplicaciones que la usan. En el comercio, finanzas, salud pública y gobierno los beneficios son sustanciales.

En esta sección se pretende proporcionar un marco para el cálculo de la inversión y su retorno, así como anticipar el nivel de detalle práctico en discusiones financieras y, finalmente, generar ideas para el análisis del ROI. Esta sección se basa en el estudio de costes realizado en el texto de Nash y otros autores titulado: *PKI: Implementing and Managing E-Security* de RSA Press. Aunque se trata de un estudio serio y completo, se realizó en el año 2001.

2.5.1. TCO: La 'I' del ROI

Se pretende desglosar los costes de la PKI al máximo y proporcionar una lista de items a considerar. Adicionalmente se deben hacer una serie de consideraciones. En primer lugar, una PKI aprovecha parte de la inversión ya realizada por la empresa, por ejemplo la propia red de comunicaciones o un directorio. Esta inversión no debe considerarse a la hora de calcular el coste de la PKI. Sin embargo, sí deben considerarse los costes si deben producirse modificaciones en la infraestructura ya adquirida, por ejemplo en el caso que deba modificarse el directorio para albergar los certificados de la PKI. Es decir, sólo deben tenerse en cuenta los costes diferenciales. En segundo lugar, la organización sólo debe considerar los items de la lista relevantes en su implantación. Es absurdo considerar costes de smart cards si no se van a usar en la implantación. En tercer y último lugar el coste no es el único criterio que debe considerarse a la hora de elegir el suministrador de la PKI. Otros criterios a considerar son la funcionalidad técnica, la fuerza financiera, la reputación, el soporte,...

Hay cuatro categorías principales en las que desglosar los costes: producto/tecnología, instalaciones físicas, personal y procesos. Los costes deben considerarse para un periodo de entre tres y cinco años para facilitar el cálculo de los presupuestos anuales.

Producto/Tecnología

La categoría producto/tecnología debe considerar los costes de todos los productos y tecnologías que constituirán la PKI, tanto si se adquieren directamente como si son proporcionados externamente. Estos costes dependen del número de usuarios y su despliegue a lo largo del tiempo. Esta categoría se divide a su vez en costes de clientes y servidores, y éstos a su vez en costes de hardware y software. Finalmente, tanto dentro del software como del hardware se deben considerar los costes relativos a adquisición, mantenimiento, actualizaciones y soporte. Algunas implantaciones deberán considerar la compra de smart cards, lectores para las smart cards, costes de su mantenimiento, actualización y soporte. Otras implantaciones simplemente deberán considerar el coste de los certificados para un número de usuarios determinado.

Instalaciones

En la categoría de instalaciones se consideran las inversiones en la planta requeridas para albergar la infraestructura de la PKI más los costes de cualquier instalación para la recuperación de desastres. Las grandes organizaciones suelen disponer de su propio centro de proceso de datos, pero si no se dispone de CPD tal vez sea necesario crearlo. En este último caso, el coste de esta última partida puede ser muy elevado.

Personal

En la categoría de personal se incluyen los costes del equipo de personas (interno, externo o ambos) involucrado en la planificación, organización, diseño, desarrollo, despliegue, formación, gestión y soporte de una PKI. Un enfoque típico suele considerar un equipo central (arquitecto, administrador de seguridad, director del proyecto) y un equipo extendido (formadores, desarroladores de software, especialistas en comunicaciones y servidores, helpdesk,...). La implicación de cada uno de los equipos varía con el desarrollo del proyecto.

Procesos

Implantar una PKI involucra muchas etapas como, por ejemplo, preparación, organización, diseño, desarrollo, despliegue, formación y subsiguiente

gestión y soporte. La PKI debe integrarse en la infraestructura existente y deben desarrollarse e implantarse los procesos operativos y de soporte. Esto es similar a cualquier otra infraestructura de la empresa.

TCO: Resumen

La clave en el cálculo del TCO es considerar todos los costes relevantes en las 4 categorías mencionadas: producto/tecnología, instalaciones, personal y procesos. La lista de elementos potenciales de coste es muy larga pero en cada caso particular se aplicarán sólo los costes relevantes y, además, sólo se deben considerar los costes diferenciales. En la tabla 2.1 está el desglose detallado de los costes.

Cuadro 2.1: Total Cost of Ownership

	Año 0	Año 1	Año 2	Año 3	Totales
Productos					
Clients					
PKI client software					
Desktop software					
Web plugins					
Maintenance/Support					
PKI client hardware					
Smartcards, tokens					
Maintenance/Support					
Subtotal					
Servers					
PKI Server software					
Certificate Server					
Security Server					
Directory services					
Authentication server					
Other functionality					
Maintenance/Support					
PKI Server Certificates					
PKI Server Hardware					
Maintenance/Support					

	Año 0	Año 1	Año 2	Año 3	Totales
Subtotal					
Planta Instalaciones Instalaciones seguras Instalaciones desastres					
Subtotal					
Personal Equipo central Director del proyecto Director seguridad Arquitecto PKI Administrador servidores PKI Administrador clientes PKI Administrador certificados PKI Equipo extendido Formadores Integradores aplicaciones Propietarios aplicaciones Especialistas redes Especialistas servidores Especialistas comunicaciones Especialistas helpdesk Especialistas desktops					
Subtotal					
Procesos Preparación Formación equipo central Validar requisitos PKI Desarrollar CP y CPS Planificación Organizar proyecto Plan operaciones y soporte Plan piloto y despliegue Plan de comunicaciones Diseño Definir certificados					

	Año 0	Año 1	Año 2	Año 3	Totales
Arquitectura Servidores PKI					
Arquitectura cliente					
Integración aplicaciones					
Administración usuarios y sop.					
Comunicaciones					
Evaluación de tecnologías					
Desarrollo					
Evaluar infraestructura IT					
Instalar servidores PKI					
Desarrollo aplicaciones					
Integración con otros sistemas					
Construir base datos PKI					
Instalar y probar aplicaciones					
Desarrollar probar adm. usu.					
Desarrollar probar adm. serv.					
Desarrollar comunicaciones					
Preparar helpdesk y soporte					
Despliegue					
Piloto - Equipo central					
Piloto - Grupo extendido					
Piloto - Grupo usuarios					
Despliegue fase 1					
Despliegue fase 2					
Despliegue fase n					
Gestión					
Helpdesk					
Administración PKI					
Administración IT					
Administración aplicaciones					
Subtotal					

2.5.2. Retorno financiero: la 'R' del ROI

En esta sección se pretende proporcionar un esquema general para estudiar el retorno financiero que provocan las aplicaciones que usan la PKI. La aproximación que vamos a aplicar es muy usada tradicionalmente cuando se pretende estudiar el valor de los procesos de negocio. Al considerar el esquema debemos tener en cuenta los siguientes puntos:

- *Focalización en los procesos de negocio*. Una PKI es una infraestructura y, como tal, el retorno financiero es nulo si no hay procesos de negocio que la utilicen. ¿De qué sirve invertir en teléfonos, faxes, sistemas de correo electrónico,... si no se usan? El retorno financiero de la PKI no puede separarse del retorno de los mismos procesos de negocio. El foco principal debe estar sobre el retorno financiero de la implementación exitosa de la seguridad de cada uno de los procesos de negocio. El retorno financiero es típicamente específico de cada aplicación, compañía o industria.

- *Establecer una métrica apropiada*. Una vez centrados en los procesos de negocio seguros, el paso siguiente es establecer métricas apropiadas para determinar el retorno financiero potencial. La métrica será una función no sólo del proceso de negocio analizado (proceso interno, proceso de cara a cliente, proceso de cara a proveedor,...), sino también del objetivo de negocio (reducción de costes, aumento de los ingresos, aumento de la eficiencia,...).

- *Medición del estado actual*. Una vez establecido un conjunto de medidas, se trata de medir con ellas el estado actual de los procesos de negocio.

- *Comparar con el estado futuro deseado*. Las mismas métricas pueden usarse para calcular el impacto financiero del nuevo o mejorado proceso de negocio.

Esta aproximación para calcular el retorno financiero no es exclusiva de las PKIs. De hecho, se aplica el mismo sistema para cualquier otra inversión significativa. Lo que se necesita para estudiar una PKI es un esquema general para organizar el estudio del retorno financiero potencial. En vistas a ello, se van a clasificar los distintos procesos de negocios afectados por las PKIs y

después se van a estudiar posibles formas de medir el retorno. Finalmente, las secciones restantes explorarán las cuatro categorías que hacen el retorno financiero posible: ingresos, costes, conformidad y riesgos.

Procesos de negocio

Dado los innumerables procesos de negocio que se pueden beneficiar de la seguridad que proporciona una PKI, es conveniente considerar un modelo más sencillo que los clasifica en unas pocas categorías. Inicialmente se pueden considerar las 3 clases siguientes de procesos de negocio: internos, orientados a cliente y orientados a proveedores. Hay que decir que ésta es una clasificación y que si bien no incluye todos los procesos de negocio, sí incluye a muchos de ellos. Otro tipo de ordenación clasifica los procesos de negocio en cuanto a su funcionalidad en: colaborativos, informativos, transaccionales, trasnferencia de fondos, distribución y relación.

Colaborativos Un ejemplo de proceso colaborativo sería el de diseño de algún producto a través de una relación cliente-proveedor. En este caso la protección de la propiedad intelectual es un requisito de seguridad de primera magnitud. Hay que asegurar que los planes del nuevo producto o las listas de clientes no se revelan a los competidores. La autenticidad de usuarios, canales de comunicaciones cifrados, acceso basado en identidad y la capacidad de proteger la información en origen o destino son requisitos de seguridad críticos. Un ejemplo de aplicación colaborativa es el correo electrónico.

Informativos Para aplicaciones orientadas a cliente, esta categoría incluye los procesos que tradicionalmente proporcionan información sobre el producto al cliente y las que permiten hacer pedidos online. Para aplicaciones orientadas a proveedores estas aplicaciones pueden incluir información sobre inventarios, planificaciones,.... La intranet corporativa, que es mayoritariamente informativa, es un típico ejemplo de aplicación interna en esta categoría.

Las aplicaciones orientadas al exterior tienen repercusión en la reputación y las ventas de la compañía. Es por ello que la autenticidad e integridad de la información son importantes. En tanto en cuanto una compañía trata con información valiosa los servicios que ofrece una PKI pueden ser de gran utilidad.

Transaccionales Abrir cuentas corrientes, presentar pedidos, seguir estado de los pedidos son ejemplos de aplicaciones transaccionales orientadas a cliente. Ejemplos de aplicaciones transaccionales orientadas a proveedor son la transmisión de órdenes de compras, transmisión de facturas,.... La seguridad de las aplicaciones transaccionales incluye la autenticidad de ambas partes de la transacción, la privacidad e integridad de los datos transmitidos, la capacidad de autorizar a usuarios a realizar ciertas transacciones basadas en su identidad y la autenticidad y no-repudio de las propias transacciones.

Transferencia de fondos Las aplicaciones de transferencia de fondos son las que tradicionalmente se han identificado con los requisitos de seguridad. El volumen de los pagos electrónicos está creciendo rápidamente debido a la demanda de las partes involucradas en dichas operaciones. En primer lugar el coste de las transacciones electrónicas es mucho menor que el coste de las mismas transacciones en papel. Además, los pagos electrónicos aportan más información sobre clientes en términos de preferencias y patrones de compra que los bancos y comerciantes pueden explotar. Esto último ha provocado la aparición de legislación para regular las compañías de servicios financieros que recogen información sobre sus clientes. En general, tanto el robo o fraude como la privacidad del consumidor fomentan el uso de soluciones de seguridad basadas en PKI para este tipo de aplicaciones.

Distribución Para bienes digitales, que pueden copiarse infinitas veces con un coste casi nulo, la distribución electrónica puede ser una bendición, ya que reduce los costes de distribución y copia, o una pesadilla si se pretende evitar la piratería y proteger la propiedad intelectual.

Relación La legislación relativa a protección de datos personales relativos a sanidad u otras áreas actúa como un estímulo significativo de la demanda de PKIs.

Finalmente, hay dos factores más que pueden tener su impacto en el retorno financiero. Por un lado la integración de sistemas y por otro la adecuación estratégica. La integración de sistemas se refiere al grado en el cual las aplicaciones internas, orientadas a cliente y orientadas a proveedor están integradas entre ellas. Un elevado grado de integración implica un elevado retorno

finaciero. La adecuación estratégica se refiere al alineamiento de los procesos de negocio de una organización con los de otras organizaciones (clientes, socios-proveedores, usuarios,...).

Con esta categorización básica podemos formular los elementos clave de la seguridad de una aplicación. El paso siguiente es establecer un conjunto de medidas para determinar el retorno financiero potencial.

Medidas

Las medidas más apropiadas son una función del proceso de negocio bajo análisis y uno o más objetivos específicos de negocio. En la tabla 2.2 se lista un conjunto de métricas potenciales para diferentes objetivos de negocio y se plantea una serie de preguntas que establecen una comparación entre el estado actual y el estado futuro. Cuantificar la respuesta a estas preguntas permite evaluar el retorno financiero de las aplicaciones PKI.

Cuadro 2.2: Medidas del retorno financiero

Proceso de negocio	Ejemplo de objetivo	Métrica potencial	Preguntas ejemplo
Orientado a cliente	Maximimizar ingresos de clientes actuales	% ingresos generados online, % de clientes existentes que operan online, % del total que el cliente gasta online, % de clientes que no completan la transacción online	Dos tercios de los clientes no completan las transacciones que requieren imprimir, firmar y enviar correo tradicional. ¿Cuál sería el impacto financiero si se redujese esta tasa a un tercio mediante el uso de firmas electrónicas, eliminación de papel y sellos y uso de e-mail?

Proceso de negocio	Ejemplo de objetivo	Métrica potencial	Preguntas ejemplo
	Minimizar costes de encontrar y adquirir nuevos clientes	% de nuevos clientes adquiridos online, Coste de la adquisición de nuevos clientes, Percepción de la marca, Conciencia de la marca	¿Cuál sería el impacto financiero si pudiéramos pasar el 50 % de los clientes online de una línea de negocio a otra línea de negocio?
	Maximizar la satisfacción del cliente, reducir helpdesk y costes de soporte	# de incidentes de órdenes incorrectas, Niveles de servicio # de servicio/peticiones helpdesk, % de servicio/ peticiones helpdesk resueltas online	¿Cuál sería el impacto financiero si el propio cliente pudiese resolver el problema online antes que a través de algún agente telefónico?
Interno	Aumentar la receptividad a las condiciones cambiantes del mercado	Ciclo orden-entrega, ciclo producto-mercado, ciclo cambio de producto	¿Cuál sería el impacto financiero si pudiéramos reducir el tiempo de algún proceso de X días a Y horas manteniendo la integridad y autenticidad de los documentos y transacciones?

Proceso de negocio	Ejemplo de objetivo	Métrica potencial	Preguntas ejemplo
	Reducir costes, aumentar la productividad	Coste de materiales, coste de servicios, productividad por empleado, % reducción helpdesk	¿Cuál sería el impacto financiero si pudiéramos aumentar la productividad por empleado y reducir las llamadas a helpdesk causadas por resets de passwords usando autenticación PKI?
Orientado a socio	Reforzar el grado de integración de sistemas con socios estratégicos	% de bienes obtenidos online, % de mantenimiento, reparaciones, suministros operativos obtenidos online	¿Cuál sería el retorno financiero si pudiéramos acortar los tiempos de entrega y reducir el inventario permitiendo que usuarios autorizados obtuvieran el 80 % del mantenimiento, reparaciones y suministros operativos mediante un navegador o un teléfono móvil?

Proceso de negocio	Ejemplo de objetivo	Métrica potencial	Preguntas ejemplo
	Reducir los costes de la asociación, mejorar la fiabilidad de los socios	Comparación de precios, coste de la conexión con socios, reparaciones y devoluciones a los socios	¿Cuál sería el retorno financiero si pudiéramos autorizar a algunos socios estratégicos con un aumento del acceso a información sensible sin comprometer la seguridad?

Como se dijo anteriormente, el retorno financiero producido por las aplicaciones PKI cae dentro de una de las siguientes cuatro categorías: ingresos, costes, conformidad y riesgos. A continuación vamos a explorar estas categorías con más detalle.

Ingresos

Los procesos de negocio que generan nuevos ingresos o un aumento de éstos suelen ser la razón más convincente para que se invierta en una PKI. Sin embargo, el incremento de ingresos por esa vía es difícil de cuantificar.

Basados en las medidas de la tabla 2.2 podemos cuantificar cualquier incremento en los ingresos de la aplicación PKI. Por ejemplo, supongamos que dos tercios de los clientes online acaban abandonando transacciones que requieren imprimir, firmar y enviar documentos en papel por correo convencional. Si esta tasa pudiese reducirse a un tercio usando firmas digitales para completar la transacción inmediatamente al mismo tiempo que se minimiza el riesgo de repudio, ¿cuál sería el incremento en los ingresos? Para muchas empresas que usan documentos intensivamente (servicios financieros, seguros,...) esto tendría un gran impacto en los ingresos por no mencionar la reducción de costes asociados a papel, impresión, servicios postales y el procesamiento tradicional de los documentos en papel.

Otra posibilidad de retorno financiero basado en ingresos incluiría el aumento de ventas a los clientes actuales y las ventas generadas a nuevos clientes. Otros ejemplos en esta categoría más difíciles de cuantificar son los producidos por la ventaja competitiva, posicionamiento estratégico y mejora de la marca o imagen corporativa.

Costes

La reducción de costes es tal vez la causa más fiable de retorno financiero de las aplicaciones PKI. Suelen ser también los retornos más fáciles de calcular. Los retornos financieros basados en costes se expresan típicamente como una combinación de:

- *Reducción de costes*. El nuevo proceso de negocio es menos costoso.
- *Evitación de costes*. El nuevo proceso de negocio escala fácilmente lo que permite ahorrar el dinero que gastaríamos en expandir el antiguo proceso de negocio.
- *Eficiencia*. El nuevo proceso de negocio ahorra tiempo, lo que permite aumentar la velocidad del e-negocio.
- *Eficacia*. El nuevo proceso de negocio aumenta la productividad. Podemos hacer más o cosas diferentes con los recursos que ya tenemos.

En la actualidad hay tres áreas particularmente fructíferas en el retorno financiero basado en costes: helpdesk, telecomunicaciones y costes asociados con el procesamiento de formularios y registros electrónicos. Los usuarios finales pueden experimentar un mejor servicio de helpdesk al mismo tiempo que se reducen los costes. Aplicaciones PKI que pueden experimentar reducciones en los costes del helpdesk incluyen intranets, VPNs y extranets seguras. Por ejemplo, una extranet segura puede proporcionar acceso a información financiera y de clientes sin interrupción 24 horas los 365 días del año. El mismo servicio ofrecido a través de agentes telefónicos sería un 40 % más caro. En cuanto a las telecomunicaciones, las VPNs junto con una autenticación fuerte no sólo mejoran la seguridad sino que pueden suponer un ahorro en los costes de helpdesk (un elevado porcentaje de llamadas de helpdesk se refieren a passwords perdidas). Por último, en empresas con un uso de documentos intensivo puede ser muy importante el ahorro en papel, impresiones,

correo postal y procesamiento de documentos. El coste de procesamiento manual de los documentos es muy elevado (copia, almacenamiento, distribución postal, recuperar documentos erróneamente almacenados, personal,...).

Conformidad

Por conformidad se entiende algunos procesos de negocio que es necesario implementar o algunos requisitos de e-seguridad que se deben cumplir. La conformidad se refiere a cosas sobre las que apenas se tiene posibilidad de elección y que se deben cumplir para estar en el negocio. El retorno financiero que provoca puede ser el ahorrar multas o bien en proteger una fuente de ingresos. En cuanto a e-seguridad los argumentos de conformidad se clasifican en las siguientes categorías:

- *Conformidad regulatoria*. Si no se cumplen pueden provocar multas, sanciones o pérdidas de ingresos.
- *Conformidad con socio*. Si no se cumplen pueden provocar perder la oportunidad de participar con un socio o grupo de socios y perder oportunidades de negocio.
- *Conformidad con cliente*. Un cliente importante puede imponer ciertas reglas a sus proveedores.
- *Conformidad para competitividad*. Si no se cumplen podemos perder una ventaja competitiva y probablemente perder ingresos.

Los casos de negocios basados en conformidad no tienen que ver tanto con un retorno financiero como con un coste del negocio o para evitar ciertas consecuencias.

Riesgos

Los argumentos basados en riesgos son usados frecuentemente para justificar inversiones en la estructura de seguridad. Campañas de marketing y ejemplos de negocio se basan en argumentos de miedo, incerteza y duda. Vender seguridad mediante el miedo puede ser efectivo hasta un cierto punto pero también margina la seguridad a un gasto operativo que puede ser recortado en la primera reducción presupuestaria. En la actualidad se pone menos énfasis en el miedo y más en el manejo sistemático del riesgo.

Las inversiones que se realizan con la prevención en mente no son visibles a menos que haya un problema, lo cual tiende a hacer la justificación basada en riesgos la menos atractiva. Por otro lado, las inversiones encaminadas a mitigar riesgos deberían focalizarse en cosas que vale la pena proteger tales como información o transacciones de alto valor. Por ejemplo: información que genera ingresos, información sobre el funcionamiento de la compañía, planes de nuevos productos, planes de marketing, investigación, o información protegida por la ley: datos del personal, datos de estudiantes, datos de pacientes,.... Una vez se ha identificado la información de alto valor se debe intentar cuantificar el impacto que los diferentes riesgos ocasionan. Por ejemplo:

- *Pérdida de productividad*. Una brecha en la seguridad puede proporcionar una interrupción sostenida de los procesos internos y de las comunicaciones.

- *Pérdida monetaria*. Una corrupción relacionada con la seguridad del sistema de gestión puede provocar retrasos en la facturación o bien un desvío de fondos.

- *Pérdida indirecta*. Un fallo en la seguridad puede provocar la pérdida potencial de ventas, pérdida de competitividad, publicidad negativa, pérdida de confianza,...

- *Riesgos legales*. Se pueden producir pérdidas relacionadas con la seguridad debidas a incumplimientos de contrato, inadecuada protección de datos,...

Evaluar el impacto financiero de los diferentes riesgos es complejo, pero las implicaciones pueden ser importantes. Los riesgos son reales. No pasa un mes sin que se publicite una brecha de seguridad si bien la mayoría pasan inadvertidas y los accesos no autorizados desde el interior de las empresas o desde el exterior son habituales.

Retorno financiero: Resumen

Los puntos más importantes para evaluar el retorno financiero de las aplicaciones PKI consisten en focalizar el proceso de negocio, establecer métricas

apropiadas y buscar los retornos relevantes en las siguientes categorías: ingresos, costes, conformidad y riesgos.

Como se ha visto en la tabla 2.2, mediante un esquema adecuado se pueden detectar los factores claves de seguridad en un proceso de negocio y comenzar a cuantificar el retorno financiero usando una técnica ampliamente aceptada.

Lo que queda es comparar la inversión realizada en el análisis del TCO con la suma de retornos financieros originados por las diferentes aplicaciones PKI.

2.5.3. PKI ROI: Resumen

¿Cuál es el retorno de la inversión de una PKI? Depende. En este apartado se han generado unos patrones que ayudan a proporcionar una respuesta más concreta. Por un lado se ha desarrollado una trama en la que capturar los diversos elementos que constituyen el coste de una implantación de una PKI. Contabiliza los costes de los productos, instalaciones, personal y procesos. Por otro lado, se ha usado una metodología ampliamente aceptada para cuantificar el retorno financiero potencial de un proceso de negocio habilitado por una PKI, incluyendo los retornos relevantes producidos por un aumento de los ingresos, reducción de los costes, mayor conformidad y evitación de riesgos. Finalmente, la comparación de las dos mitades de la ecuación del ROI en cada empresa u organización en particular determinará el ROI propio de la PKI en concreto. En general, se puede decir que los beneficios de una PKI compensan ampliamente los costes.

A título de ejemplo el Bank of Bermuda disponía de una infraestructura que permitía el acceso remoto de clientes a servicios del banco a través de la red telefónica a cualquiera de los 14 puntos de presencia internacional del banco. Sin embargo, particularmente para clientes situados en lugares donde el banco no tenía presencia la solución era cara y poco fiable (un cliente en Sudáfrica tenía que conectarse con Londres). La implantación de la PKI tuvo el beneficio inmediato de permitir la conexión a través de Internet y cualquier cliente con un navegador tenía conexión con los servicios a través de una conexión cifrada. Otros beneficios fueron el volumen de negocio que permitió alcanzar la PKI, reducción del tiempo de proceso,...

Como se comentó al inicio de esta sección, el presente estudio de costes se basa en el texto de Nash del año 2001. No obstante, en estos últimos 10 años

se ha evolucionado mucho y actualmente se propone otro tipo de soluciones para implantar una PKI. Para empresas pequeñas o incluso de un tamaño respetable los costes del despliegue de una PKI son inasumibles. En la práctica lo que se hace es externalizar el servicio de manera que otra empresa especializada en PKIs se encarga de gestionarlo. Un servicio externalizado reduce la complejidad y asegura la escalabilidad y la disponibilidad. Una PKI externalizada reduce drásticamente los costes de despliegue. Se comparan 3 grandes áreas del coste de una solución PKI: software, infraestructura y personal.

Una solución in-house requiere pagar las licencias del software, su mantenimiento y soporte. Si, por el contrario, se externaliza el servicio habría una tasa correspondiente al despliegue inicial y luego unas tasas anuales. El soporte, las licencias y el mantenimiento están incluidos en esas tasas. Además la recuperación de desastres puede ya estar incorporada. Teniendo en cuenta todas estas consideraciones, el coste software de una solución in-house es más elevado que el de una solución externalizada.

En cuanto a la infraestructura, la solución externalizada puede permitir ahorrar no sólo todos los costes relativos a su adquisición y mantenimiento, sino también el esfuerzo de IT para instalar y gestionar la infraestructura.

Otra fuente importante de ahorro es el personal. La tecnología PKI requiere personal especializado. Los consultores y el personal de IT necesitará implantar los componentes hardware y software, crear y hacer cumplir las políticas, gestionar el ciclo de vida de los certificados y crear un plan de recuperación de desastres para una solución in-house. En cambio, una solución externalizada sólo requiere un técnico IT a tiempo parcial para ocuparse de las tareas relacionadas con la PKI. La reducción de costes es enorme.

2.6. Construir o comprar

Una organización puede decidir construir su propia PKI desde los cimientos o bien comprar los productos y servicios PKI ya sea insource o outsource. La mayoría de las organizaciones optan por comprar la PKI debido a varias razones:

- La tecnología PKI es relativamente compleja. La mayoría de los vendedores que ofrecen software PKI han invertido mucho en tecnología y el retorno de la inversión sólo es posible a través de múltiples ventas.

Una organización que construya totalmente su PKI difícilmente podrá recuperar la inversión.

- Dada la complejidad del software es improbable que la organización disponga de los recursos necesarios para el desarrollo.
- Las patentes tienen un impacto en el coste del desarrollo.

Debido a todas estas dificultades la mayoría de organizaciones descartan la opción de construir su propia PKI.

2.7. Insource o Outsource

Una vez se ha dedidido comprar una PKI hay tres opciones para implantarla: insource, outsource o una aproximación híbrida. Esta decisión debe tomarse antes de embarcarse en el diseño e implementación de la solución PKI. Esta decisión puede afectar al ámbito, costes y la elección de posibles vendedores.

Las soluciones insource las instala, implementa, administra y mantiene la propia organización. El departamento de IT de la organización debe liderar la implantación de la tecnología PKI: hardware CA, base de datos CA, directorio PKI y los enlaces de comunicaciones entre todas las entidades participantes.

Las soluciones outsource delegan la mayoría de funciones sobre otra organización. El grado de outsourcing puede variar desde la simple generación de algunos certificados para servidores hasta la total externalización de múltiples servicios de CA dedicados en exclusiva a la propia empresa.

Las soluciones híbridas combinan outsourcing y insourcing. La propia organización mantiene parte de las CAs y otra compañía mantiene las otras CAs. Este modelo es adecuado para muchas compañías. Una empresa puede tener diferentes necesidades de seguridad para diferentes aplicaciones. Aplicaciones con altos requisitos de seguridad pueden tener sus propias CAs implementadas y administradas por la misma compañía; la implementación y gestión de CAs usadas para aplicaciones con bajos requisitos de seguridad puede ser externalizada.

Un factor decisivo cuando se toma la decisión de externalizar o no la PKI es el nivel de control y flexibilidad que una compañía requiere en sus CAs. Generalmente la solución insource proporciona una flexibilidad máxima ya que las CAs son gestionadas por la propia empresa. Además, los procesos y

mecanismos de autenticación pueden ser completamente definidos por la organización. Esta flexibilidad es crítica cuando se trata de definir documentos legales como las Certificate Policy (CP) y los Certificate Practice Statements (CPS). Un modelo outsource proporciona menos flexibilidad ya que la jerarquía de certificados se crea cuando se instalan las CAs. Cambios adicionales requieren más tiempo y dinero en un modelo outsource. Si una organización no tiene unos requisitos específicos de seguridad estará dispuesta a aceptar los CP y CPS de la empresa externa y esos documentos serán mantenidos por la empresa externa. Además, la tarea de proteger las claves privadas de las CAs recae sobre la empresa externa.

El modelo externalizado permite periodos de despliegue más cortos debido a que los requisitos de instalación y gestión de hardware y software son menores. Además, en el modelo externalizado los requisitos de seguridad e infraestructura decrecen ya que la CA soporta esa responsabilidad. En un modelo externalizado los costes se distribuyen en un periodo de tiempo mayor, frente a un mayor desembolso inicial en un modelo insource. Para organizaciones grandes los modelos insource pueden ser convenientes ya que la infraestructura de seguridad y el personal pueden estar ya disponibles. Los fuertes gastos iniciales pueden verse compensados al proporcionar costes más reducidos a lo largo del tiempo debido a las economías de escala. Además, una organización grande tendrá probablemente unas necesidades específicas de seguridad y autenticación. La desventaja de los modelos insource es que requieren mayores tiempos de despliegue.

2.8. Entorno cerrado o abierto

En un entorno cerrado las comunicaciones son siempre dentro del mismo dominio. Puede estar formado por una o varias organizaciones operando con los mismos procedimientos y restricciones. En un entorno abierto las comunicaciones pueden ser entre diferentes dominios y los procedimientos serán diferentes.

Un entorno cerrado puede tener productos de diferentes fabricantes por lo que es necesario evitar soluciones propietarias que dificulten la interoperabilidad sobre todo si la comunidad de propietarios crece con el tiempo. En el caso de entornos abiertos es claro que la interoperabilidad es una cuestión

Cuadro 2.3: Comparación insource vs outsource

Área	Insource	Outsource
Políticas	Políticas flexibles	Políticas poco flexibles
Customización	Máxima flexibilidad y tiempo de despliegue mayor	Mínima flexibilidad y tiempo despliegue menor
Despliegue	Requiere mayor esfuerzo inicial pero menor esfuerzo subsiguiente	Esfuerzo menor
Coste	Moderado coste inicial y menor coste subsiguiente	Menor coste inicial y moderado subsiguiente
Personal	Número significativo para mantener y gestionar	Menor número para mantener y gestionar

crucial. En cualquier caso la tecnología seleccionada debe basarse en estándares y el vendedor debe comprometerse a ofrecer productos compatibles con otros fabricantes.

2.9. Aplicaciones específicas o solución global

Es posible usar muchos servicios de seguridad sin implantar una PKI. Se trata típicamente de soluciones puntuales enmarcadas en una única aplicación. La mayoría de organizaciones grandes no están interesadas en soluciones puntuales y prefieren una infraestructura de seguridad global que cumpla con los requisitos de múltiples aplicaciones y, por tanto, sea mucho más efectiva en términos de coste. Sin embargo, en organizaciones pequeñas y cerradas las soluciones puntuales pueden ser un método barato de cumplir los requisitos de seguridad. El producto Lotus Notes, por ejemplo, ofrece correo electrónico, bases de datos documentales y la posiblidad de automatizar flujos de documentos conjuntamente con firmas digitales y cifrado.

2.10. Selección del producto

Después de haber evaluado los requisitos de seguridad presentes y futuros de la empresa, después de haber decidido qué aplicaciones son susceptibles de usar la PKI y tras haber considerado la externalización o no del servicio se debe decidir el producto o productos que se implantarán. Es necesario tener en cuenta los siguientes factores a la hora de seleccionar el producto:

- *Coste*. ¿Cuánto costará la solución PKI? ¿Qué recursos requerirá para administrarla y mantenerla?
- *Facilidad de uso*. ¿Es difícil solicitar un certificado? ¿Cuánto se tarda en conseguir un certificado? ¿Cuánto se tarda en renovar un certificado? ¿Es compleja la administración?
- *Tiempo*. ¿Cuánto se tarda en poner la PKI en funcionamiento? ¿Cuánto se tarda en adaptar una aplicación a la PKI?
- *Soporte de estándares*. ¿Está la solución PKI basada en estándares? Los estándares son clave para la interoperabilidad.
- *Gestión PKI*. ¿Cuáles son las características de gestión de una solución dada? ¿Qué características quiere la organización?

La decisión sobre externalizar o no tiene un impacto definitivo en la elección del producto. Si se selecciona una solución outsource hay que elegir entre las CAs comerciales. Insource permite elegir entre diferentes vendedores de software PKI.

2.11. Desarrollo de una PKI

En esta sección se pretende esbozar el guión que se debería seguir en la implantación de una PKI.

El primer paso en el desarrollo de una PKI consiste en comprender los objetivos de negocio y los requisitos que provocan la necesidad de la implantación de una PKI. Conviene revisar los procesos de negocio y proponer una arquitectura para el sistema. El documento SP (Security Policy, Política de Seguridad) de la organización debe ser una importante fuente de información

durante esta etapa. Con toda esta información se define el ámbito y otros parámetros significativos de la PKI.

Seguidamente se deben analizar las aplicaciones y los datos que la organización debe proteger y evaluar las amenazas y riesgos a los que están sujetos. Se debe considerar el impacto de un compromiso de la seguridad; por ejemplo, el perjuicio que podría provocar la revelación de información confidencial a un competidor o el acceso incontrolado a datos financieros. En ocasiones las leyes obligan a proteger los datos de los clientes y, en tal caso, una brecha en la seguridad puede implicar sanciones y multas. La redacción de una CP sin haber planteado previamente estas cuestiones provocará ineludiblemente el fracaso del proyecto. La CP también debe adecuarse a los datos y aplicaciones que debe proteger. Si la CP no es apropiada será demasiado cara o dejará datos críticos desprotegidos. Hay una correlación entre el nivel de seguridad y el coste de la PKI. Una CP muy segura requerirá más personal y procedimientos más complejos que una CP menos exigente.

Algunas veces puede ser necesario desarrollar más de un documento CP, particularmente cuando las transacciones que se desean proteger tienen requisitos de seguridad muy diferentes. Pensar, por ejemplo, en una organización que deba mantener, por un lado, una VPN para vendedores que facturen pequeños importes y, por otro, transacciones financieras de millones de euros llevadas a cabo por los contables de la compañía. En estos casos la organización tiene dos opciones: implementar dos políticas diferentes o soportar ambos tipos de transacciones con la política más estricta. La compañía debe comparar los costes y beneficios de ambas estrategias y tomar una decisión. Esta información también puede usarse para analizar los costes y beneficios de un seguro de responsabilidad. La compañía puede contratar una póliza de seguro para cubrir las responsabilidades que resulten de errores del personal de la PKI.

Antes de redactar el documento CP conviene proveerse de una copia de documentos CP de otras organizaciones con objetivos similares. También es útil revisar los distintos apartados en los que se divide el documento CP y determinar qué puntos son aplicables en nuestro entorno. En general estos documentos son fáciles de obtener ya que las organizaciones suelen publicarlos.

Si se prevé que la PKI se va a cross-certificar con PKIs de otras organizaciones, se deben investigar los perfiles de certificados y CRLs que usan dichas PKIs. Hacer el perfil de nuestros certificados y CRLs consistente con los de

las otras organizaciones reducirá los problemas de compatibilidad posteriormente. Se deben considerar aspectos como las extensiones críticas, algoritmos y los métodos para localizar información de revocación.

Con todas estas consideraciones ya se está en condiciones de redactar uno o varios documentos CPs adecuados para nuestra organización. En el ejemplo presentado anteriormente de la compañía que debe proveer seguridad para contables y vendedores, los contables pueden usar módulos criptográficos hardware validados contra *FIPS 140-1 Level 2* o superiores, mientras que el personal de ventas tendría suficiente con módulos criptográficos software validados contra *FIPS 140-1 Level 1*. El documento CP puede requerir una identificación presencial cuando un contable obtiene sus credenciales, mientras que un correo certificado puede ser suficiente para autenticar un vendedor.

El paso siguiente sería decidir si la propia organización opera la PKI o, simplemente, la subcontrata. Para operar la PKI la compañía necesita una instalación segura para sus operaciones y una instalación remota para almacenar backups. Si la organización no dispone de estas instalaciones, su construcción puede ser muy costosa. En ocasiones puede ser rentable realizar algunas operaciones de la PKI localmente y subcontratar las restantes. Por ejemplo, las RAs y el directorio pueden operarse localmente, mientras que las operaciones de CA pueden ser externalizadas. Si se decide operar la PKI, a continuación se deben seleccionar los productos que pueden usarse para implementar la política. Diferentes productos PKI se diseñan con diferentes objetivos. La cuestión clave es seleccionar productos que implementen la política de seguridad redactada en el documento CP. Si los productos no soportan los objetivos del documento CP, se deberán diseñar procedimientos complejos para soslayar los inconvenientes de los productos con el consiguiente encarecimiento del mantenimiento de la PKI.

Los estudios de costes se hallan estrechamente ligados con los pasos anteriores. Se pueden realizar varios estudios de costes comparando diferentes productos y comparando la operación interna de la PKI con la externalización. Todo este proceso puede requerir varias pasadas hasta conseguir diseñar una solución PKI adecuada.

El paso final de la documentación consiste en redactar los documentos CPSs. Si la propia organización va a operar las CAs, los CPSs deben indicar la ubicación que se usará. Los CPSs además especifican cómo se usan los controles proporcionados por los productos PKI para cumplir los objetivos descritos en el documento CP. Si, por el contrario, se externalizan los servi-

cios, el proveedor del servicio ya tendrá probablemente escrito el documento CPS y debe demostrar que sus controles y procedimientos cumplen con la CP.

A continuación, ya se está en condiciones de desplegar físicamente la PKI en la organización. Conviene comenzar por adiestrar el personal y realizar unas pruebas piloto con un pequeño conjunto de usuarios. A menudo, estas experiencias iniciales resultan en revisiones de los documentos CPSs. Limitando el conjunto inicial de usuarios se consigue que la mayoría de usuarios se beneficie de su experiencia. Una vez se ha conseguido el documento CPS definitivo ya se puede inciar el despliegue masivo de la PKI.

Capítulo 3

Diseño y despliegue

3.1. Introducción

Una vez evaluada la necesidad de implantar una PKI en la organización, se puede pasar al diseño de la solución propiamente dicho. De hecho se puede considerar que la redacción de los documentos CP y CPS (tratada en el capítulo anterior) forma ya parte del diseño de la PKI, si bien no desde un punto de vista tecnológico.

Es imposible abarcar la pluralidad de situaciones que se dan en el mundo real por lo que en este capítulo se darán ejemplos y descripciones generales de cuestiones tecnológicas relativas al diseño y despliegue de una PKI. Más concretamente se estudiarán las distintas arquitecturas PKI existentes, el diseño de certificados, el diseño de una PKI jerárquica, ejemplos de uso de extensiones, cuestiones de interoperabilidad, impacto en la infraestructura, integración en el directorio, software cliente, APIs...

3.2. Arquitecturas PKI

La arquitectura de una PKI describe la organización de sus CAs y sus relaciones de confianza. Cada arquitectura tiene sus ventajas y sus inconvenientes y es apropiada para algunos entornos, mientras que para otros no lo es. En esta sección vamos a analizar las diferentes arquitecturas, desde las más sencillas a las más complejas. Las cuatro primeras arquitecturas son aplicables a una organización, mientras que las tres últimas se refieren a las arquitecturas de la interconexión de organizaciones.

3.2.1. CA única

La arquitectura PKI más básica es la formada por una CA única que proporciona todos los certificados y CRLs para una comunidad de usuarios. Todos los usuarios confían en la CA que emitió su propio certificado. Por definición, no pueden añadirse nuevas CAs a la PKI y puesto que sólo hay una única CA no se establecen relaciones de confianza con otras CA. Es la arquitectura más simple de implementar. Los caminos de certificación constan de un único certificado y hay una única CRL. Por contra, esta arquitectura no es escalable y presenta un único punto de fallo. Si se compromete la CA se invalidan todos los certificados emitidos. Cada usuario debe ser informado inmediatamente. Para restablecer la confianza se debe volver a emitir todos los certificados y la información sobre el nuevo punto de confianza debe ser distribuida a todos los usuarios. Esta arquitectura sólo es aplicable a una empresa que no necesita comunicarse con el mundo exterior. La figura 3.1 muestra una CA única.

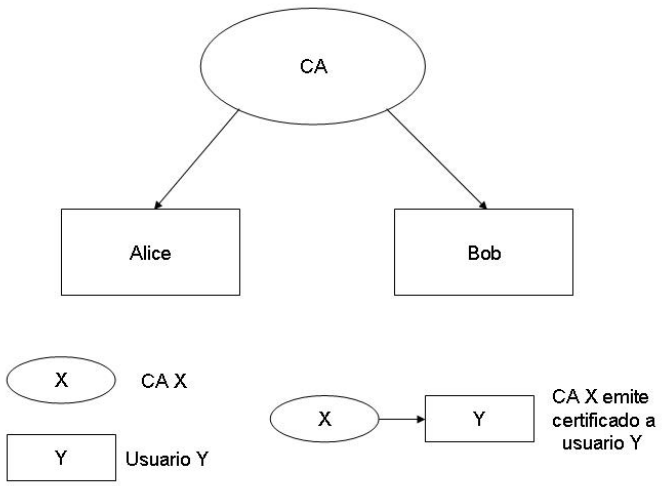

Figura 3.1: CA única

3.2.2. Listas de confianza simple

La *lista de confianza* es la forma más simple de soportar más de una CA. En esta arquitectura hay más de una CA pero no hay relaciones de confianza entre ellas. En este modelo cada usuario mantiene una lista de las CAs en las que confía. Se pueden añadir nuevas CAs a la PKI modificando las listas de los usuarios. Los usuarios aceptan certificados y CRLs emitidos por una CA en su lista de confianza. Un usuario requiere un certificado y una CRL. La construcción y validación de caminos de certificados es muy simple.

La principal ventaja de esta arquitectura es su simplicidad. Los caminos de certificados constan de un único certificado y es fácil añadir una nueva CA a la PKI. Sin embargo, hay importantes desventajas. La nueva CA que se desea añadir a la lista debe investigarse previamente. Si se compromete una CA, probablemente se informará rápidamente a sus propios usuarios, pero no a los usuarios de otras CAs ya que la CA comprometida no tiene manera de saber qué usuarios la tienen en su lista. Un mecanismo muy similar es usado en los navegadores Web. El vendedor del navegador lo preconfigura con un conjunto extenso de certificados CA raíz conocidos. Esta característica facilita el uso del navegador, ya que la mayoría de certificados de servidores Web son emitidos por CAs bien conocidas y, por tanto, son automáticamente reconocidos. Muchos navegadores no se han diseñado para verificar la validez de los certificados, como consecuencia una CA comprometida puede seguir usándose por muchos usuarios antes de que la lista sea actualizada o, todavía peor, se distribuya una nueva versión de la aplicación. La figura 3.2 muestra la arquitectura de una lista de confianza simple.

3.2.3. Jerárquica

La arquitectura jerárquica es la más tradicional. En esta arquitectura varias CAs, con una relación superior-subordinado, proporcionan servicios a la PKI. En esta arquitectura todos los usuarios confían en la *CA raíz*. Con la excepción de la CA raíz, todas las CAs tienen una única CA superior. Una CA puede emitir certificados a CAs, usuarios o ambos. Cada relación de confianza entre CAs se representa por un único certificado. El emisor es la CA superior y el sujeto es la CA subordinada. Para añadir una nueva CA a la PKI, una CA existente emite un certificado a la nueva CA. La nueva CA se injerta bajo la CA existente y se convierte en una CA subordinada de la CA emisora. Dos PKI jerárquicas pueden fusionarse de la misma manera. Las CAs superio-

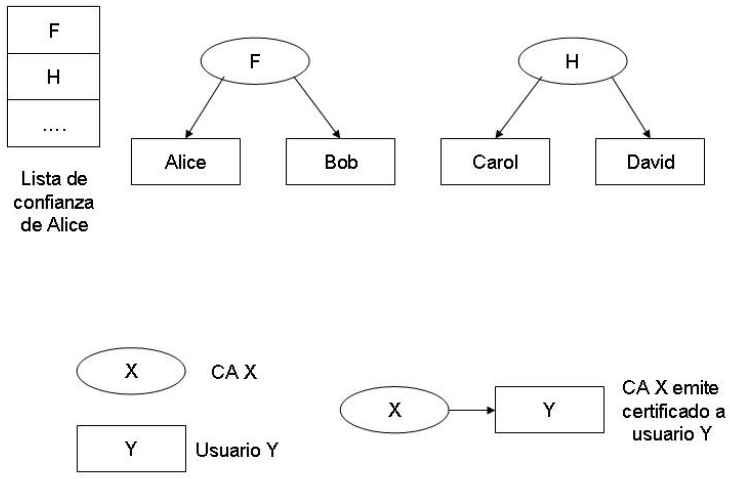

Figura 3.2: Lista de confianza simple

res pueden imponer restricciones a las CA subordinadas. Estas restricciones pueden implementarse con procedimientos o en los propios certificados.

La arquitectura jerárquica proporciona varias ventajas que contribuyen a hacer de ella uno de los modelos más ampliamente desplegados hasta la fecha. En primer lugar, los caminos de certificados son relativamente cortos. Además, puesto que todos los usuarios confían en la misma CA raíz, hay un único camino para alcanzar un usuario específico. Esto permite a la entidad final (usuario) distribuir los certificados de cualquier CA intermedia en la cadena junto con su propio certificado y dar el camino al usuario del certificado. La desventaja más significativa de este modelo radica en la misma razón que su simplicidad y éxito: la existencia de una CA raíz en la que todos confían. En una comunidad pequeña es posible acordar una única CA raíz, pero en comunidades grandes es imposible que todos acuerden una única CA raíz.

Si se compromete una CA (diferente de la raíz), su CA superior simplemente revoca su certificado. Una vez se ha restablecido, la CA emite certificados a todos sus usuarios. La CA superior emite un nuevo certificado con la

clave pública de la nueva CA insertándola de nuevo en la jerarquía. Durante el periodo de restablecimiento, dos usuarios que no pertenecen a la parte comprometida de la jerarquía pueden seguir operando. El compromiso de la CA raíz tiene el mismo impacto que en la arquitectura de CA única. Hay que informar a todos los usuarios del compromiso, restablecer la CA, emitir todos los certificados y distribuir el nuevo punto de confianza. Sin embargo, todavía hay una ventaja en comparación con el compromiso de la CA única y es que la CA raíz debe emitir un número mucho menor de certificados por lo que puede operar mucho tiempo offline, reduciendo la posibilidad de un compromiso.

El modelo jerárquico funciona razonablemente bien dentro de los confines de una empresa, particularmente si la empresa tiene una estructura organizativa fuertemente jerárquica. El modelo tiende a dejar de funcionar cuando se atraviesan los confines de la organización. Las razones son usualmente la falta de acuerdo sobre una CA raíz única y las diferentes políticas operativas instituidas en las diferentes organizaciones.

La figura 3.3 es una representación gráfica de una PKI jerárquica.

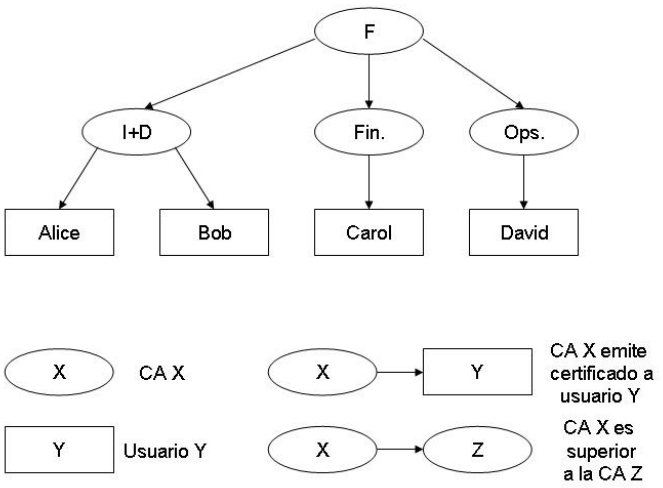

Figura 3.3: PKI jerárquica

3.2.4. Malla

La arquitectura en malla es la principal alternativa a la PKI jerárquica. En este modelo varias CAs proporcionan los servicios PKI, pero la relación entre ellas es de igual a igual y no jerárquica. Cada usuario confía en una única CA; sin embargo, no es la misma CA para todos los usuarios. En general, los usuarios confiarán en la CA que emitió su certificado. Las CAs emiten certificados entre ellas, un par de certificados describe una relación de confianza bidireccional. Una nueva CA se añade a la malla simplemente intercambiando certificados con otra CA que ya es miembro de la malla.

La construcción de caminos es particularmente complicada en una malla. El proceso para construir caminos en esta arquitectura no es determinista, conlleva múltiples elecciones (algunas elecciones llevan a un camino válido y otras no), puede producir lazos y, en general, los caminos son más largos que en una PKI jerárquica. Los problemas que aparecen al determinar un camino son similares a los problemas de encaminamiento en una red de routers de Internet al enviar un paquete entre terminales de la red. El aumento en la flexibilidad que se deriva de usar una malla tiene su contrapartida en el aumento de la complejidad para formar caminos. Otro inconveniente de las mallas es que una CA puede incluir en la malla otra CA que sea un competidor mío y con el que no me interese tener ninguna relación de confianza. Esto refuerza la necesidad de mecanismos para controlar este tipo de situaciones. La localización de certificados en las mallas es otra cuestión importante. A diferencia de la arquitectura jerárquica, no es posible predeterminar los caminos asociados con cada entidad final. La construcción de caminos en una malla depende fuertemente de la existencia y fácil acceso a directorios para localizar certificados.

Las arquitecturas en malla son muy flexibles. El compromiso de una CA no provoca la completa inutilización de la PKI. Las CAs que emitieron certificados a la CA comprometida simplemente los revocan, lo que conlleva la eliminación de la CA en la PKI. En el mejor de los casos, la PKI se reduce en una CA y sus usuarios. En el peor de los casos la PKI se fragmenta en PKIs más pequeñas.

La figura 3.4 representa gráficamente una arquitectura en malla.

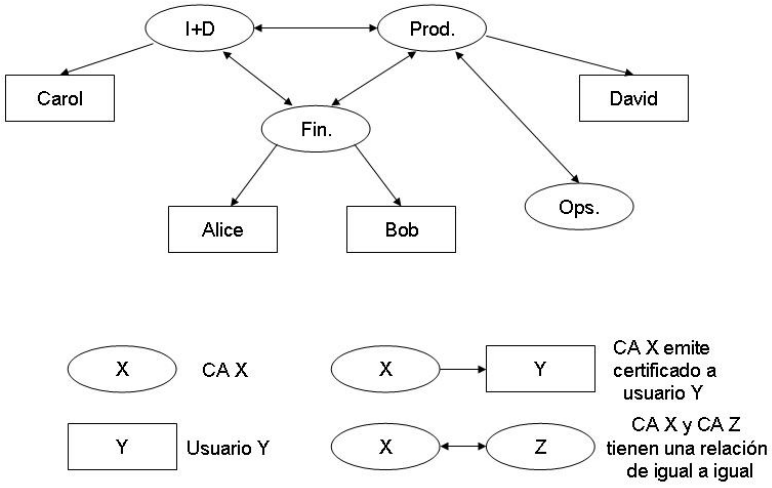

Figura 3.4: PKI en malla

3.2.5. Lista de confianza extendida

Esta arquitectura, a diferencia de las anteriores, permite extender la PKI a varias empresas u organizaciones. La arquitectura lista de confianza extendida corrige los defectos de la lista de confianza simple. Cada usuario mantiene una lista de puntos de confianza. Cada punto de confianza identifica una PKI en la que el usuario confía y cuya arquitectura puede ser CA única, jerárquica o malla. En esta arquitectura, el usuario añade una CA por cada PKI en la que confía.

Esta arquitectura conserva la ventaja esencial de la lista de confianza simple: la facilidad y rapidez con la que un usuario puede confiar en otras PKIs. Además, también cuenta con la principal ventaja de las PKI en malla y jerárquica que consiste en que al confiar en una CA extiende su confianza a otras CAs relacionadas con la anterior, con lo que se reduce el número de puntos de confianza que se debe mantener en la lista. No obstante, los problemas relacionados con el mantenimiento de listas extensas de puntos de confianza y con el compromiso de CAs persisten. Esta arquitectura también introduce

sus propios inconvenientes: la construcción de caminos de certificados es más compleja ya que el usuario no sabe cuál de las CAs en las que confía le llevará al certificado del usuario. La construcción de caminos suele hacerse partiendo del certificado del usuario hasta llegar a una de las CAs en las que confía.

La figura 3.5 muestra una lista de confianza extendida.

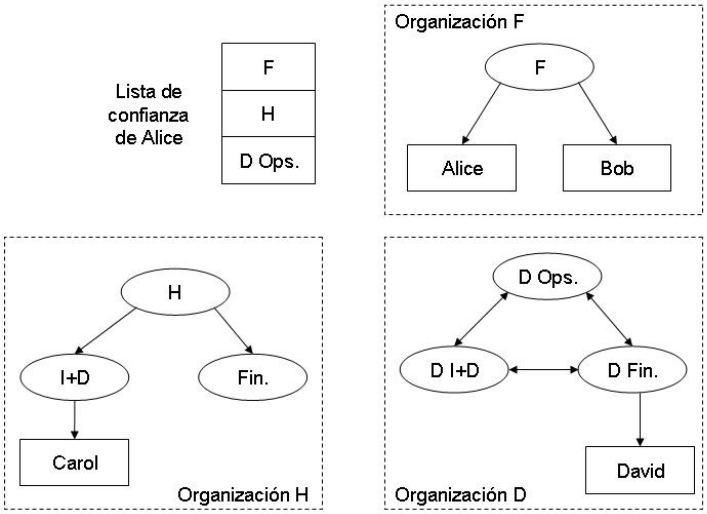

Figura 3.5: PKI Lista de confianza extendida

3.2.6. PKIs Cross-certificadas

Si dos empresas o comunidades de usuarios tienen unos requisitos crecientes de comunicaciones seguras, las PKIs pueden establecer relaciones de confianza de igual a igual. Cada usuario puede tener un único punto de confianza. Las empresas se cross-certifican de igual a igual, mientras que dentro de una empresa las relaciones pueden ser jerárquicas o en malla.

Un usuario no puede añadir una CA por su propia iniciativa. Los administradores de las CAs revisan las políticas y prácticas de la otra CA antes de cross-certificarse y están más cualificados que un usuario para decidir si la

otra PKI es confiable. Una cross-certificación permite que todos los usuarios mantengan comunicaciones seguras. Con las listas de confianza extendidas cada usuario debe actualizar su propia lista.

La construcción de caminos de certificados puede ser bastante difícil porque la PKI resultante combina secciones en malla y jerárquicas. Por otro lado, los certificados pueden ser bastante complejos. Sin embargo, puesto que todas las PKIs están directamente conectadas entre ellas, el compromiso de una CA es rápidamente comunicado a las otras PKIs que no tienen más que revocar el certificado adecuado. El problema más importante de esta arquitectura es que el número de certificados crece rápidamente cuando el número de PKIs que hay que interconectar aumenta.

La figura 3.6 muestra una PKI cross-certificada.

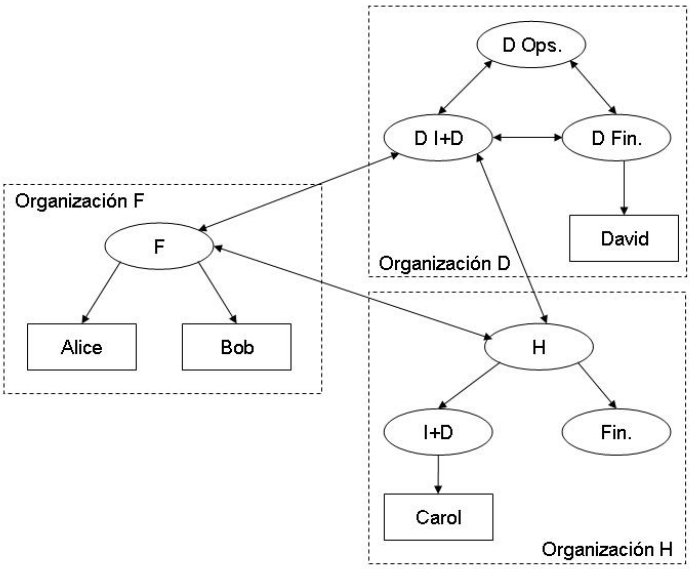

Figura 3.6: PKI cross-certificadas

3.2.7. Arquitectura de CA puente

Por un lado, no se puede esperar que un usuario mantenga una lista de confianza extensa; por otro lado, los administradores de CAs precisan un me-

canismo más eficiente para establecer relaciones de confianza con otras PKIs que la cross-certificación. La arquitectura de CA puente está diseñada para soslayar esos problemas.

La CA puente no emite certificados a los usuarios ni se constituye en un punto de confianza para éstos. La CA puente establece relaciones de confianza de igual a igual con las PKIs de las diferentes organizaciones o empresas. Si una empresa tiene una PKI jerárquica, la CA puente establecerá una relación de confianza con la CA raíz. Si, en cambio, la empresa presenta una PKI en malla, la CA puente establecerá una relación con una de sus CAs. Ningún usuario configura la CA puente como un punto de confianza. Es simple añadir PKIs a una PKI de CA puente; para ello basta con establecer una relación de confianza entre la nueva PKI y la CA puente. El número de certificados crece con el número de PKIs interconectadas, por lo que no crece tan rápidamente como en una PKI cross-certificada. La CA puente no resuelve el problema de construcción y validación de caminos, sigue siendo tan complejo como en una PKI en malla. Finalmente, la PKI se puede recuperar fácilmente del compromiso de una CA, basta con que la CA puente revoque el certificado que emitió a la PKI comprometida, el resto de relaciones no se alteran. Si es la propia CA puente la que se ve comprometida, ésta lo notifica al resto de PKIs, las cuales a su vez revocan el certificado que emitieron a la CA puente. Esta, a su vez, revoca los certificados que emitió. El resultado es un conjunto de PKIs separadas de tal manera que los usuarios de diferentes PKIs no serán capaces de establecer relaciones seguras entre ellas. Sin embargo, es fácil crear de nuevo una CA puente.

La figura 3.7 muestra una arquitectura de CA puente.

3.2.8. Elección de la arquitectura

Ninguna arquitectura es perfecta. Cada una de ellas presenta fortalezas y debilidades. La elección depende de los requisitos de la organización.

Una única CA es la solución más sensata para una comunidad de usuarios pequeña si se consigue un acuerdo sobre la CA. Los problemas asociados al desarrollo de caminos de certificados y su validación desaparecen. Pero, en cambio, el compromiso de la CA es catastrófico, si bien su reconstrucción puede ser relativamente sencilla.

Una PKI jerárquica es la solución más elegante con una estructura bien definida y se deriva de una forma natural de la propia jerarquía de la organi-

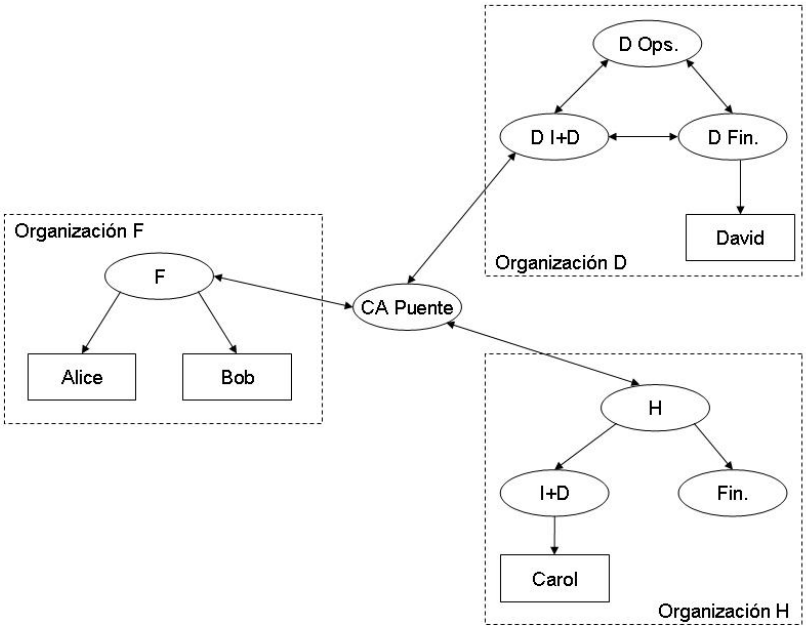

Figura 3.7: PKI de CA puente

zación. La construcción y validación de caminos de certificados son sencillas. Pero puede ser difícil imponer esta estructura a una organización, particularmente si ya se ha desplegado una colección de CAs independientes. Si una empresa pretende desplegar una PKI jerárquica, es mejor desplegar la CA raíz en primer lugar e integrar en la jerarquía las restantes PKIs a medida que son desplegadas. El compromiso de la CA raíz es, de nuevo, una catástrofe, aunque con las adecuadas medidas de protección es altamente improbable. El compromiso de otra CA es más sencillo de resolver.

Una PKI en malla es una solución práctica para organizaciones que no poseen una estructura bien definida. Si se han establecido CAs de antemano en la organización, la PKI en malla es la solución más directa. El compromiso de una CA es catastrófico para sus usuarios, pero las transacciones para los usuarios de las otras CAs no se ven afectadas. La complejidad de la construcción y validación de caminos de certificación es su gran inconveniente.

Cuando la cross-certificación de PKIs no es posible, las listas de confianza extendida son la solución. A pesar de sus limitaciones, las listas de confianza

extendida trasladan las decisiones a los usuarios. A causa de que muchas organizaciones no han desplegado PKIs todavía, estas listas de confianza son la solución más extendida.

La cross-certificación de PKIs es una estructura sencilla para un reducido número de organizaciones. Sin embargo, deja de ser práctica cuando el número de partes crece o cuando las relaciones entre las organizaciones son dinámicas.

Las CAs puente son adecuadas para conectar un gran número de PKIs y también si las relaciones son dinámicas. Las compañías pueden establecer y terminar relaciones rápidamente.

3.3. Diseño de certificados y CRLs

Esta sección es un ejemplo de diseño de perfil de certificados y CRLs. Para ello se realizan una serie de asunciones que modelizan un entorno imaginario. No obstante, el entorno y las sugerencias son lo suficientemente generales como para adaptarse a muchas situaciones reales. Ejemplos de otros perfiles de certificados y CRLs ya existentes son PKIX para Internet, SET para soportar transacciones de pago con tarjeta de crédito sobre Internet o el *US Federal PKI X.509 Certificate and CRL Extensions Profile*. A lo largo del ejemplo se observa cómo el entorno condiciona algunos campos de certificados y CRLs y los criterios que se aplican para definir el contenido de otros campos. En el ejemplo se consideran esencialmente CRLs y certificados de usuarios. Esta sección se basa en el artículo de Sharon Boeyen titulado *X.509 Profiles for various CA scenarios*. Las consideraciones que se detallan a continuación deben aparecer en la sección de perfiles de certificados y CRLs de CP y CPS.

El entorno que sive de ejemplo tiene las siguientes características:

- El repositorio PKI es un directorio LDAP.
- Se emiten certificados a CAs y usuarios.
- El mecanismo de revocación usado son CRLs.
- Las CRLs pueden particionarse por tipo de certificado (usuario / CA) y número de certificado, pero no por la razón de la revocación.
- No se usan CRLs indirectas ni CRLs deltas.

- Los certificados se pueden aplicar a varias aplicaciones.
- Los usuarios tienen dos pares de claves (firma y cifrado).
- El identificador de política *any police* debe estar inhibido.

Antes de abordar el contenido propiamente dicho de los campos se detallan unas recomendaciones sobre las funciones hash a utilizar y el tamaño de las claves basadas en el documento *SP 800-57: Recommendation for Key Management - Part 1* del NIST de Marzo del 2007. Según este documento las claves públicas de usuarios y de CAs que no vayan a usarse más allá del año 2030 deben ser como mínimo de 2048 bits de longitud si se utiliza RSA y de 224 bits si se usa criptografía de curvas elípticas. Las firmas creadas con estas claves deben usar la función hash SHA-256. Para claves públicas de CAs y usuarios que vayan a ser usadas más allá del año 2030 la recomendación sugiere una longitud mínima de 3072 bits para RSA y 256 bits para curvas elípticas. De nuevo se recomienda usar la función SHA-256 para firmas creadas con dichas claves. Si se desean claves más largas se debería usar la función hash SHA-384 con claves RSA de 7680 bits y claves de curvas elípticas de 384 bits. La función hash SHA-512 es apropiada para claves RSA de 15360 bits de longitud y claves de curvas elípticas de 512 bits.

3.3.1. Recomendaciones comunes de certificados

Las recomendaciones de esta subsección son comunes a todos los perfiles de certificados (usuario, CA subordinada, CA crosscertificada,...).

Campos básicos

Los estándares RFC5280 y X.509 ordenan la inclusión de la mayoría de los campos básicos en todos los tipos de certificados.

Version Hasta la fecha ha habido 3 versiones de sintaxis de los certificados X.509. La única versión que permite incluir extensiones es la versión 3. Las extensiones juegan un papel clave en la fiabilidad y adaptación de un certificado a una determinada aplicación, por ello todos los certificados en la actualidad deben ser de la versión 3.

Serial Number El estándar X.509 obliga a que el valor de este campo sea único para cada certificado emitido por una CA dada. La combinación de los campos *Issuer Name* y *Serial Number* proporciona un identificador único para cada certificado, siempre y cuando el *Issuer Name* sea único. Se recomienda que los valores del campo *Serial Number* comiencen con el entero 1 y se incrementen subsiguientemente de uno en uno. De este modo se facilita la interoperabilidad entre sistemas.

Signature Algorithm Este campo contiene el identificador del algoritmo usado por la CA para firmar el certificado y debe coincidir con el campo del mismo nombre de la porción no firmada del certificado. Por motivos de interoperabilidad el algoritmo más recomendado es RSA con la función hash apropiada para el nivel de seguridad. En la actualidad los certificados deberían ser firmados por la función hash SHA-256 y el algoritmo RSA PKCS #1 Version 1.5 con claves públicas de CAs de 3072 bits de longitud. El OID para estas firmas es *sha256WithRSAEncryption* iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-1(1) 11. Otro algoritmo que puede usarse en aplicaciones que requieran mayor seguridad es RSA-PSS. Si lo que se busca es mejorar el rendimiento y altos niveles de seguridad se sugiere usar el algoritmo de firma ECDSA basado en curvas elípticas.

Issuer Name y Subject Name Estos campos contienen el DN (Distinguished Name) del emisor y sujeto del certificado. Un DN es sintácticamente una secuencia de pares atributo y valor. Cada atributo de cada par está identificado por un OID conocido, mientras que la porción valor es una cadena de caracteres. Hay dos esquemas generalmente usados para formular DNs: el esquema geopolítico y el esquema de nombres de dominio.

El esquema geopolítico formula DNs basados en la localización geográfica de la entidad junto con atributos del usuario o la organización. Por ejemplo, una organización ubicada en Estados Unidos podría nombrar su CA como *ou = ABC CA; o = ABC; c = us*. En algunos países hay autoridades de registro que gestionan los nombres de las organizaciones. Estas organizaciones garantizan la unicidad del nombre y permiten resolver disputas entre organizaciones sobre el nombre. Este esquema de nombres es aplicable si existe una infraestructura de directorio X.500 o LDAP. Si se usa este esquema de nombres se recomienda que el DN de una CA contenga como mínimo los atributos de *countryName*, *organizationName* y *organizationalUnitName*. Para usuarios

finales se recomienda que el DN incluya además el atributo *commonName*.

El esquema de nombres de dominio basa los DNs en el estándar RFC2247 que proporciona un mapeo de nombres de dominio a DNs usando el atributo *domainComponent* (dc) para representar cada componente de un DN. Este esquema es aconsejable en entornos en los que el nombre de dominio es suficiente para identificar la organización y no existe un directorio LDAP o X.500. Por ejemplo una organización podría nombrar su CA como *cn = ABC CA; dc = ABC; dc = com*.

Se recomienda usar el esquema de nombres de dominio ya que no se requiere registrar adicionalmente el nombre de la organización y no hay ambigüedades en los nombres.

Validity Period Consta de los campos *notBefore* y *notAfter*. La recomendación X.509 permite codificar dichos campos con los formatos *utcTime* y *generalizedTime*, pero las recomendaciones PKIX exigen que las fechas hasta el año 2049 se codifiquen como *utcTime* y para el año 2050 y posteriores se use *generalizedTime*. La principal diferencia entre los dos formatos es que el formato *utcTime* representa los años con 2 dígitos, mientras que *generalizedTime* usa 4 dígitos. En la actualidad y dado que los certificados tienen una duración breve se usa el formato *utcTime* expresado como GMT (Zulú) asegurando de esta forma la independencia de la ubicación física de las aplicaciones PKIX que usan los certificados. Si un certificado se modifica durante su periodo de validez debe ser revocado. Para compensar el tiempo que dichos certificados han de permanecer en las CRLs se recomienda que los certificados de usuarios tengan una duración de 2 años, los certificados de CAs subordinadas o CAs cross-certificadas tengan una duración de 5 años y los certificados de la raíz jerárquica tengan una duración de 10 años.

Subject Public Key Info Este campo transporta la clave pública y el algoritmo para usar la clave. Por interoperabilidad el algoritmo recomendado es *rsaEncryption* cuyo OID es *iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-1(1) 1*. Este OID se usa para certificados de clave pública tanto para firmar como para cifrar. La aplicación concreta de la clave se especifica en la extensión *key usage*. Algunas aplicaciones con mayores requisitos de seguridad pueden usar el algoritmo *RSA-PSS* para firmar y el algoritmo *RSA-OAEP* para cifrar. Si lo que se quiere es mejorar el rendimiento se recomienda usar algoritmos de curvas elípticas. Para certificados que usen RSA se reco-

mienda tamaños de claves públicas de 2048 bits y para los que usen curvas elípticas se recomiendan tamaños de 224 bits.

Issuer y Subject Unique Identifiers Son campos opcionales en la recomendación X.509. Se añadieron en la versión 2 previamente a las extensiones. Son campos raramente usados y su uso no está recomendado.

Extensiones

Las extensiones más comunes están descritas en las recomendaciones X.509 y PKIX. Hay algunas extensiones que deben incluirse en todos los certificados y otras que sólo deben incluirse en algún tipo de certificados. A continuación se detallan las extensiones que deben incluirse en todos los certificados.

Key Identifiers Una CA puede tener más de una clave pública y el campo *authorityKeyIdentifier* es un puntero al certificado que contiene la clave pública que verifica la firma del certificado. Esta extensión puede estar formada por el *Issuer Name* y el *Serial Number* o bien por una función hash de la clave pública del certificado de la CA que firma. En la actualidad se recomienda usar la función hash SHA-256 y se desaconseja el uso de los campos *Issuer Name* y *Serial Number*. Esta extensión debe incluirse en todos los certificados.

La extensión *subjectKeyIdentifier* identifica la clave pública certificada en el certificado y sirve para localizar rápidamente una clave pública entre múltiples claves. Esta extensión debe incluirse en todos los certificados. El contenido del campo suele ser la función hash SHA-256 de la clave pública certificada.

Estas extensiones se usan para crear cadenas de certificados en la construcción de caminos de certificados.

Basic Constraints La extensión *basicConstraints* identifica el sujeto del certificado como un usuario o una CA. Es una propiedad importante de cada certificado cuando se construyen los caminos de certificación. Por ello se recomienda su uso tanto en certificados de usuario como de CA.

Key Usage La extensión *keyUsage* identifica los propósitos para los cuales se va a usar la clave certificada. En todos los certificados de CA esta extensión debe estar presente con el bit *keyCertSign* activado. En muchos casos las

CAs también emiten CRLs, en tal caso el bit *cRLSign* también debe activarse. También debe usarse esta extensión en certificados de usuario. En tal caso el bit *keyEncipherment* significa que la clave pública se usa para cifrar claves secretas para ser distribuidas y el bit *digitalSignature* significa que la clave certificada se va a usar para firmar documentos o mensajes.

Certificate Policies La extensión *certificatePolicies* indica las políticas bajo las cuales debe usarse el certificado. Las políticas indican un conjunto de aplicaciones o un nivel de seguridad asociado con el certificado. Se recomienda la inclusión de esta extensión en todos los certificados, si bien el indicador *anyPolicy* se debe evitar.

Authority Info Access La extensión *authorityInfoAccess* tiene dos usos primordiales. Por un lado apunta a la ubicación del repositorio donde se hallan los certificados de la CA emisora. Se recomienda que esta extensión incluya una URI LDAP para ayudar en la construcción de caminos de validación. Por otro lado esta extensión proporciona punteros a servidores OCSP.

CRL Distribution Points La extensión *cRLDistributionPoints* apunta a la ubicación de las CRLs que listarán el certificado en caso de que éste fuera revocado. Puesto que se ha asumido el uso de CRLs en esta sección se recomienda el uso de esta extensión y que su contenido sea una URI LDAP.

Criticidad

Por motivos de interoperabilidad se recomienda que la mayoría de extensiones no sean críticas. Sin embargo, debido a su vital importancia se recomienda que las extensiones *basicConstraints* y *keyUsage* sean críticas en los certificados de CAs y *keyUsage* sea crítica en los certificados de usuarios.

3.3.2. Contenido de CRLs

Las CRLs consisten en un conjunto de campos básicos y un conjunto de extensiones. Algunas extensiones aplican a la CRL completa y otras a un único certificado en la lista.

Campos básicos de CRLs

Version Hay 2 versiones de CRLs en la recomendación X.509. La versión 2 es la única que permite incluir extensiones. Debido al papel primordial que juegan las extensiones todas las CRLs deben ser de la versión 2.

Signature Este campo contiene el OID del algoritmo usado por la CA para firmar la CRL y debe coincidir con el campo del mismo nombre de la porción no firmada de la CRL. Por motivos de interoperabilidad el algoritmo más recomendado es RSA con la función hash apropiada para el nivel de seguridad. En la actualidad los certificados deberían ser firmados por la función hash SHA-256 y el algoritmo RSA PKCS #1 Version 1.5 con claves públicas de CAs de 3072 bits de longitud. El OID para estas firmas es *sha256WithRSAEncryption* iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-1(1) 11. Otro algoritmo que puede usarse en aplicaciones que requieran mayor seguridad es RSA-PSS. Si lo que se busca es mejorar el rendimiento y altos niveles de seguridad se sugiere usar el algoritmo de firma ECDSA basado en curvas elípticas.

Issuer Este campo contiene el DN del emisor de la CRL. Los estándares permiten las CRLs indirectas donde una entidad diferente a la emisora del certificado emite la CRL. No obstante, en general, es la misma CA que emite los certificados la que emite las CRLs. Debido a las asunciones que hemos realizado, este campo coincide con el campo *Issuer* de los certificados.

This Update y Next Update Son válidas las mismas recomendaciones que para el campo *Validity Period* de los certificados por lo que se usa el formato *utcTime* con la fecha y hora GMT (Zulú). El campo *Next Update* es opcional según la recomendación X.509, pero obligatorio según el perfil PKIX. Este campo juega un papel fundamental cuando se validan certificados ya que sirve para determinar si la CRL que estamos utilizando es apropiada para ser usada en el instante actual o si ya hay disponible una CRL más reciente.

Revoked Certificates Este campo contiene la lista de los certificados revocados. Cada entrada de la lista incluye el número de serie del certificado revocado y la fecha y hora de la revocación. Para la codificación de la fecha y

hora de revocación se aplican las mismas directrices que para el campo *Validity Period* de los certificados, es decir el formato *utcTime* con la fecha y hora GMT (Zulú). Además el campo *revokedCertificates* puede incluir una o más extensiones aplicables al certificado en la componente *crlEntryExtensions*. Si la causa de la revocación es conocida debe incluirse en la extensión *reasonCode*. Si la razón de revocación es un compromiso de la clave, debe incluirse la extensión *invalidityDate* porque si bien esta información no tiene impacto en las validaciones que se llevan a cabo en la actualidad, sí puede tener impacto en validaciones futuras realizadas contra un tiempo de interés en el pasado. La extensión *certificateIssuer* no debe incluirse ya que es el mismo que el emisor de la CRL. La extensión *holdInstructionCode* tiene un soporte reducido y no debe incluirse.

Extensiones de CRLs

Según las asunciones que se han comentado al inicio de esta sección, hay 2 extensiones aplicables a la totalidad de la CRL que deben incluirse siempre.

Authority Key Identifier Esta extensión identifica la clave pública que se debe usar para verificar la firma de la CRL. Se recomienda usar la función hash SHA-256 de la clave pública usada para firmar, mientras que se desaconseja el uso del *Issuer Name* conjuntamente con el *Serial Number*.

Issuing Distribution Point Esta extensión incluye varios campos. El campo *distributionPoint* indica la entrada del directorio en la que se halla la CRL y debe coincidir con el contenido de la extensión *cRLDistributionPoints* de los certificados. Este campo está pensado para que un agente de la PKI, que puede muy bien ser una RA, sepa dónde publicar una CRL emitida por una CA. También se debe incluir en esta extensión el campo *onlyContainsCACerts* o bien el campo *onlyContainsUserCerts*. Los campos *onlySomeReasons* y *indirectCRL* no deben usarse.

3.3.3. Certificados de usuarios

En la tabla 3.1 se muestran las recomendaciones para un certificado de usuario. Muchas recomendaciones se basan en interoperabilidad y amplia

aplicabilidad. La columna P muestra si el campo o extensión debe estar presente (p), ausente (a) o bien es opcional (o). La columna C indica si la extensión es crítica (c) o no crítica (nc). El siguiente perfil muestra recomendaciones.

Cuadro 3.1: Certificado de usuario

Campo/Extensión	P	C	Valor/Comentarios
Campos básicos			
version	p	n/a	2 (indica certificado v3)
serialNumber	p	n/a	entero monótono creciente empezando por 1
signature	p	n/a	*algorithm* presente e incluye el OID 1.2.840.113549.1.1.11 *parameters* presente con valor NULL
issuer	p	n/a	El DN de la CA codificado en *PrintableString*. Los atributos incluyen *domainComponent* (dc) y *commonName* (cn)
validity	p	n/a	*notBefore* instante de emisión del certificado *notAfter* 2 años después de la emisión del certificado codificado en *utcTime*
subject	p	n/a	DN del usuario codificado en *PrintableString*. Los atributos incluyen *domainComponent* (dc) y *commonName* (cn)
subjectPublicKey Info	p	n/a	La componente *algorithm* de *AlgorithmIdentifier* incluye el OID 1.2.840.113549.1.1.1. La componente *parameters* de *AlgorithmIdentifier* está presente con valor NULL. *subjectPublicKey* incluye una clave pública RSA de 2048 bits
issuerUnique Identifier	a	n/a	
subjectUnique Identifier	a	n/a	
Extensiones			

Campo/Extensión	P	C	Valor/Comentarios
authorityKey Identifier	p	nc	*keyIdentifier* incluye los 256 bits de la función hash SHA-256 de la clave pública usada para verificar la firma del certificado. *authorityCertIssuer* y *authorityCertSerialNumber* se excluyen.
subjectKey Identifier	p	nc	Incluye los 256 bits de la función hash SHA-256 de la componente *subjectPublicKey* del campo *subjectPublicKeyInfo*
authorityInfo Access	p	nc	*caIssuers* está presente e incluye la URI LDAP que apunta a la entrada de directorio de la CA emisora
basicConstaints	p	nc	*cA* falso y *pathLenConstraint* se excluye
keyUsage	p	c	El bit *digitalSignature* se habilita para certificados de verificación. El bit *keyEncipherment* se habilita para certificados de cifrado.
cRLDistribution Points	p	nc	*distributionPoint* está presente y apunta con una URI LDAP a la entrada de directorio donde se encuentra la CRL. Se excluyen *reasons* y *cRLIssuer*
certificatePolicies	p	nc	*policyIdentifier* incluye al menos un OID representando las políticas bajo las cuales se emitió este certificado
subjectAltName	p	nc	Se incluye *rfc822Name*
privateKeyUsage Period	o	nc	Incluido sólo en certificados de verificación. *notBefore* es la fecha de emisión del certificado. *notAfter* es el 70 % del periodo de vida del certificado. De esta manera hay tiempo suficiente para la reemisión del certificado. Codificado como *utcTime*
extKeyUsage	o	nc	Incluido sólo si es necesario para aplicaciones específicas
subjectDirectory Attributes	a	n/a	
freshestCRL	a	n/a	

Campo/Extensión	P	C	Valor/Comentarios
policyConstraints	a	n/a	
inhibitAnyPolicy	a	n/a	
policyMappings	a	n/a	
nameConstraints	a	n/a	
issuerAltName	a	n/a	
subjectInfo Access	a	n/a	

3.3.4. CRLs

En la tabla 3.2 se muestran las recomendaciones para una CRL. Muchas recomendaciones se basan en interoperabilidad y amplia aplicabilidad. El perfil recomienda CRLs particionadas para reducir su tamaño aunque el mismo perfil se puede utilizar en entornos en los que el particionado no está soportado. La columna P muestra si el campo o extensión debe estar presente (p), ausente (a) o bien es opcional (o). La columna C indica si la extensión es crítica (c) o no crítica (nc). El siguiente perfil muestra recomendaciones.

Cuadro 3.2: CRLs

Campo/Extensión	P	C	Valor/Comentarios
Campos básicos CRL			
version	p	n/a	1 (indica CRLs v2)
signature	p	n/a	*algorithm* está presente e incluye el OID: 1.2.840.113549.1.1.11. *parameters* incluye el valor NULL
issuer	p	n/a	Incluye el DN de la CA emisora y es idéntico al campo *issuer* de los certificados emitidos por la misma CA
thisUpdate	p	n/a	El instante de emisión de la CRL en formato *utcTime*
nextUpdate	p	n/a	No mayor que 24 horas después del campo *thisUpdate* en formato *utcTime*

Campo/Extensión	P	C	Valor/Comentarios
revokedCertificates	p	n/a	Los componentes *serialNumber* y *revocationDate* en cada entrada de la lista
Extensiones para cada entrada de la lista de la CRL			
reasonCode	p	nc	Incluido salvo que la razón no esté especificada
invalidityDate	o	nc	Incluido solo si *reasonCode* está presente e indica *keyCompromise* o *cACompromise*
holdInstruction Code	a	n/a	
certificateIssuer	a	n/a	
Extensiones globales de la CRL			
authorityKey Identifier	p	nc	*keyIdentifier* está presente y contiene los 256 bits de la función hash SHA-256 de la clave pública de la CA usada para verificar la firma de la CRL. Se excluyen los campos *authorityCertIssuer* y *authorityCertSerialNumber*
cRLNumber	p	nc	entero monótono creciente que comienza por 1
issuingDistribution Point	p	c	La componente *distributionPoint* contiene el DN de la entrada de directorio que contiene la CRL. Las componentes *onlyContainsUserCerts* y *onlyContainsAuthorityCerts* se habilitan según se trate de CRL o ARL. Las componentes *onlySomeReasons* y *indirectCRL* se excluyen
issuerAltName	a	n/a	
crlScope	a	n/a	
statusReferrals	a	n/a	
cRLStream Identifier	a	n/a	
orderedList	a	n/a	
deltaInfo	a	n/a	
deltaCRLIndicator	a	n/a	

Campo/Extensión	P	C	Valor/Comentarios
baseUpdateTime	a	n/a	
freshestCRL	a	n/a	

3.4. Diseño de una CA jerárquica

En esta sección vamos a detallar el proceso de diseño de una PKI jerárquica. Antes de desplegar una PKI jerárquica la organización debe dedicar tiempo a diseñarla. Para desarrollar la estructura correcta se deben tener en cuenta los siguientes requisitos:

- Requisitos de las aplicaciones PKI.
- Requisitos de seguridad.
- Requisitos técnicos.
- Requisitos de negocio.
- Requisitos externos.

Los elementos de la jerarquía que se deben diseñar son:

- Número de niveles
- Organización de las CAs en la jerarquía
- Tipos de certificados que cada CA emitirá
- Los tipos de CAs que se desplegarán en cada nivel
- Medidas de seguridad para proteger las CAs
- Si se requieren diferentes políticas de certificados

El número de niveles de la jerarquía es una consideración básica del proceso de diseño de una jerarquía. También es necesario determinar las CAs individuales requeridas en cada nivel.

Algunas organizaciones con pocos usuarios y que requieran servicios PKI básicos tienen suficiente con una única CA raíz. En este caso la CA no se saca de la red y está siempre disponible para emitir certificados a usuarios, servicios o dispositivos de red. Es una solución con un coste mínimo y una gestión sencilla, pero, por contra, no es una solución redundante.

Una jerarquía de 2 niveles comprende una CA raíz offline y una o más CAs emisoras. Para mejorar la redundancia pueden instalarse 2 CAs emisoras con la misma configuración y si una CA falla la otra puede seguir ofreciendo el servicio. Por otro lado las CAs del segundo nivel pueden forzar el cumplimiento de diferentes políticas.

Una jerarquía de 3 niveles proporciona la mejor combinación de seguridad y flexibilidad y consiste en una CA raíz offline, una o más CAs offline para forzar el cumplimiento de políticas en el segundo nivel y una o más CAs subordinadas emisoras formando el tercer nivel. El nivel de seguridad es muy alto debido a que la CA raíz y las CAs del segundo nivel se hallan offline. La flexibilidad es muy alta ya que diferentes CAs del segundo nivel pueden forzar el cumplimiento de diferentes políticas. La gestión de las CAs puede dividirse entre diferentes grupos de administradores.

El número y ubicación geográfica de las CAs emisoras se basa en distintos criterios:

- Número de certificados que se emitirá, se necesitan más CAs para emitir más certificados.
- Configuración geográfica de la red WAN, las CAs deben distribuirse de acuerdo con la topología de la WAN.
- Modelo de gestión de la PKI, diferentes equipos de gestión administran diferentes aplicaciones PKI.
- Organigrama de la compañía, pensar en una compañía que incluye varias compañías subordinadas.
- Categorías de los empleados, se pueden crear diferentes CAs para cada categoría de empleado.
- Regulaciones, en ocasiones hay regulaciones del sector que condicionan el diseño de la jerarquía.

En muchas ocasiones el despliegue de una PKI está motivado por la introducción de una o más aplicaciones que dependen de la existencia de una PKI. Esto suele determinar quién gestionará la aplicación, el tipo de entidades a las que se emitirá certificados (usuarios, servicios, dispositivos de red,...), el número de entidades, los tipos de certificados y cómo los gestionará la aplicación.

Los requisitos de seguridad se describen en los documentos CP y CPS e incluyen, por ejemplo, alojar las CAs en salas de servidores físicamente seguras con control de acceso, protección para las claves privadas de CAs usando, por ejemplo, HSMs (Hardware Security Module) y los diferentes requisitos de emisión de los certificados.

Los requisitos técnicos pueden obligar a delegar la administración a una región u oficina específica, esto se puede conseguir desplegando una CA en dicha oficina y asignando la gestión de la CA a los usuarios de dicha oficina. Otros requisitos técnicos pueden prevenir fallos de hardware. Para ofrecer alta disponibilidad de las CAs pueden usarse clusters de servidores, de manera que cuando caiga un servidor su carga de trabajo sea absorbida por otros servidores. Para prevenir fallos en los discos suelen usarse configuraciones RAID 5 o RAID 1 de manera que un fallo en un disco no provoca la caída del servicio.

Una etapa primordial en la definición de una jerarquía es determinar los periodos de validez de los certificados en cada nivel de la jerarquía incluidos los certificados emitidos a los usuarios. La estrategia recomendada para determinar los periodos de validez es comenzar por los certificados emitidos a los usuarios. Hay que tener en cuenta que un certificado no puede tener un periodo de vida que exceda el periodo de vida del certificado de la CA emisora. Por ejemplo, si el certificado de la CA expirará en 13 meses y el periodo de vida de un certificado es de 2 años la CA emitirá certificados con un periodo de vida de 1 año. Una regla que suele usarse es hacer el periodo de validez del certificado de una CA de al menos el doble del periodo de validez de los certificados que emite. Así, por ejemplo, los periodos de validez de los certificados en una CA jerárquica de 3 niveles podrían ser de 1 ó 2 años para los certificados de los usuarios, 5 años para los certificados de las CA subordinadas y 10 años para la CA raíz.

Otro parámetro destacado en la configuracion de una jerarquía es la longitud de las claves de las distintas CAs. El compromiso de una clave privada de una CA en un nivel alto de la jerarquía tiene un impacto superior al com-

promiso de una clave privada en un nivel inferior. Por otro lado la seguridad de una clave aumenta con su longitud. Por todo ello las claves son de mayor longitud en los niveles más altos de la jerarquía. En la práctica algunas aplicaciones no soportan tamaños de claves superiores a un determinado número de bits y obligan a que todas las claves de CAs de la jerarquía no excedan dicho número de bits.

Finalmente para cada CA de la jerarquía se deben definir los puntos de publicación de CRLs y certificados o bien la localización del servidor OCSP. Los protocolos más usados son LDAP y HTTP, pero la elección depende de la frecuencia de publicación, de los protocolos que pueden atravesar los firewalls y de los propios sistemas operativos empleados.

Los requisitos de negocio definen los objetivos de la organización. El caso típico es el de una organización que desea mejorar sus procesos de negocio. Algunos requisitos de negocio que afectan el diseño de una jerarquía son:

- Minimizar costes de la PKI, se pueden combinar CAs de políticas y emisoras para reducir el número de CAs.
- Alta disponibilidad para la emisión de certificados, puede requerir clustering de servidores.
- Definición de responsabilidades de las CAs, definidas en el documento CPS.

Los requisitos externos los imponen otras organizaciones con las que se desea trabajar o los gobiernos de los países en los que se realizan negocios. Por ejemplo, una organización con la que se desea trabajar puede obligar a generar una jerarquía que defina políticas internas y externas diferentes. Los gobiernos pueden obligar a las organizaciones a gestionar la información de sus empleados o clientes de una determinada manera. Los certificados y CRLs deben publicarse en ubicaciones accesibles desde el exterior para permitir la validación de certificados desde una red externa.

3.5. Impacto en la infraestructura

La implantación de una PKI tiene un cierto impacto sobre la infraestructura existente que debe analizarse.

En primer lugar se añadirán servidores a la infraestructura existente para soportar las funciones de CAs y RAs. Los servidores de CAs necesitan instalaciones físicamente seguras.

Las aplicaciones PKI incrementarán el tráfico de red. Por un lado los usuarios deberán consultar certificados y CRLs, por otro lado los mensajes y transacciones firmados son más largos que los no firmados. Se debe evaluar el impacto que la población de usuarios de la PKI tendrá en el tráfico de red.

Si la PKI se usa para autenticación, el servidor donde se almacenan los certificados y la propia red deben ser altamente fiables. Cualquier indisponibilidad del servicio puede provocar importantes problemas. Muchas de las transacciones que llevan a cabo CAs y RAs no son tan críticas aunque en el caso de transacciones de importes muy elevados puede ser crítica la publicación de CRLs.

Las plataformas sobre las que se ejecutan los servicios PKI deben ser adecuadamente protegidas. Los servicios innecesarios deben ser detenidos, se deben cerrar los puertos de comunicaciones salvo los estrictamente necesarios, se deben de aplicar todos los parches de seguridad, se debe prevenir la emisión de certificados falsos o la generación de solicitudes de certificados falsas. En el repositorio se debe prevenir la sustitución de certificados legítimos por certificados falsos.

Los procedimientos de backup y recuperación son críticos para los servidores PKI y tienen requisitos de seguridad especiales. Por ejemplo, si un servidor CA falla el proceso de restauración debe garantizar que la base de datos de certificados es la correcta y que no ha sido alterada.

Finalmente, el personal encargado de la gestión de la PKI debe ser adecuadamente adiestrado.

3.6. Integración en el directorio

Muchas organizaciones optan por integrar el repositorio de certificados en el directorio de la propia organización.

Uno de los obstáculos es que no hay un estándar de directorio universalmente aceptado. En la actualidad LDAP parece haber ganado la batalla, pero hay también soluciones de directorios propietarias y soluciones basadas en DNS. Este amplio abanico de opciones provoca en ocasiones problemas de interoperabilidad entre las organizaciones.

Junto con el problema de los diferentes estándares, existe la cuestión de la interoperabilidad entre diferentes vendedores. No todos los vendedores implementan las mismas funciones de los estándares y pueden incluso implementarlas de forma no consistente. Sin embargo, se avanza hacia unas implementaciones cada vez más compatibles.

El despliegue de un directorio conlleva siempre cuestiones de escalabilidad y rendimiento asociadas. El número de servidores del servicio de directorio dependerá del número de certificados, del periodo de validez de las CRLs y de otras varibles dependientes de la PKI. Por otro lado es altamente improbable que el directorio esté dedicado únicamente a la PKI por lo que tendrá que soportar también una carga adicional a la de la propia PKI.

3.7. Software cliente

Se necesita un componente PKI en el lado del cliente. En la actualidad son las aplicaciones PKI individuales las que incorporan la lógica relativa a la seguridad PKI (cumplimiento de políticas, verificación de revocaciones, validación de certificados, ...) pero por motivos de consistencia y uniformidad la funcionalidad de seguridad debe colocarse fuera de la propia aplicación aunque accesible desde ésta. Este tipo de arquitectura es escalable, fácil de gestionar y refleja la definición de PKI. Incorporar la funcionalidad de seguridad en el propio sistema operativo no es una solución suficientemente general porque puede dificultar la interoperabilidad multiplataforma. A corto plazo la uniformidad y consistencia en la seguridad son más fácilmente alcanzables a través de un componente cliente PKI.

Algunos dispositivos como teléfonos móviles o PDAs (Personal Digital Assistants) no tienen la capacidad de procesamiento necesaria para incorporar la funcionalidad de un cliente PKI completo aunque sus usuarios pueden requerir los beneficios de autenticación, integridad y confidencialidad en sus comunicaciones que ofrece una PKI. En tales casos una posible solución es descargar parte de la funcionalidad sobre uno o varios servidores. Esta argumentación favorece una arquitectura con un cliente muy reducido.

De cualquier manera es necesaria la existencia de algún tipo de cliente PKI. Esta arquitectura permite a los sistemas operativos y aplicaciones desentenderse de la necesidad de comprender y procesar las cuestiones relativas a la seguridad. Además esta arquitectura facilita la administración. Por ejemplo, si

se descubre un defecto en algún protocolo de seguridad es más fácil actualizar un módulo cliente que modificar todas las aplicaciones que incorporan dicho protocolo.

El cliente PKI es un componente crítico en el diseño de una PKI y debe tenerse en cuenta cuando se planifica un despliegue en una organización con una amplia variedad de plataformas y dispositivos.

3.8. Application Programming Interfaces

Una API (Application Programming Interface) es una interfase ofrecida por una plataforma (hardware y sistema operativo) a las aplicaciones que se ejecutan sobre dicha plataforma y consiste típicamente en un conjunto de funciones. Una plataforma puede ofrecer varias APIs a las aplicaciones. Una API criptográfica (crypto API, CAPI) es un conjunto de llamadas que permiten a un programador de aplicaciones acceder a funcionalidades criptográficas.

La manera más habitual de agregar funcionalidades PKI a aplicaciones nuevas o ya existentes es usar CAPIs que ya implementan funciones PKI. El uso de CAPIs evita la implementación de funciones criptográficas en cada aplicación y aporta mejoras significativas en la seguridad. Una CAPI de alto nivel permite a un programador con pocos conocimientos criptográficos realizar una operación (por ejemplo cifrar) sin necesidad de que el programador proporcione los detalles de cómo llevarla a cabo. Una API de bajo nivel permite especificar detalles criptográficos al programador pasándolos como parámetros a la llamada.

Una cuestión crucial al diseñar o seleccionar una CAPI es cómo se llevan a cabo las funciones que implementa. Hay dos alternativas: software o hardware. En el primer caso, el software puede ser parte del sistema operativo o bien un paquete adicional que proporcione servicios criptográficos. En el caso del hardware, suele tratarse de un *Hardware Security Module* (HSM) de algún tipo. Tales dispositivos proporcionan seguridad física a las claves y adoptan muy diversas formas. Un HSM típico puede implementarse como una tarjeta que puede ser instalada en una ranura PCI de un PC, una tarjeta PCMCIA o bien como un dispositivo SCSI externo.

Las funciones encargadas de gestionar las claves (generar, importar, exportar, borrar,...) suelen ofrecerse a través de una interfase diferente más fuertemente protegida por motivos de seguridad. De este modo, tener acce-

so a las funciones de la CAPI no implica tener acceso a las claves y un código malicioso que se ejecutase en el sistema podría acceder a las funciones de la CAPI pero tendría mucho más difícil comprometer las claves del sistema. Habitualmente las claves que se almacenan en los HSMs son simplemente claves maestras que se usan para cifrar y proporcionar integridad a las claves restantes que se almacenan externamente. Estas claves almacenadas externamente se pasarán cifradas como un parámetro de la función CAPI. El módulo puede entonces descifrar y verificar la integridad de la clave antes de utilizarla para cifrar los datos. En una CAPI software el secreto e integridad de las claves reposa exclusivamente sobre las medidas de protección que puede ofrecer el sistema operativo y éstas suelen ser bastante limitadas.

Si bien no hay auténticos estándares internacionales para CAPIs, hay una serie de CAPIs que han alcanzado un amplio reconocimiento. Entre las más importantes destacan *Generic Security Service Application Program Interface* (GSS-API), publicada en RFC2743 y otros documentos de soporte, y PKCS #11 publicada por RSA. Un área estrechamente relacionada con las CAPIs son los módulos criptográficos. El estándar *NIST FIPS Pub. 140-2* especifica requisitos de seguridad para los módulos criptográficos y establece 4 niveles de seguridad. Estos niveles tienen implicaciones en el diseño de la CAPI correspondiente al módulo relativos al control de acceso a la CAPI (nivel 2) y a la separación de la gestión de claves y la CAPI (nivel 3 y superiores).

La especificación GSS-API define servicios y primitivas a un nivel independiente del mecanismo criptográfico subyacente y del entorno del lenguaje de programación. Esto permite que los servicios GSS-API se hayan implementado mediante tecnologías de clave secreta y también con tecnologías de clave pública y que el código fuente de las aplicaciones que hacen llamadas a esta API se pueda portar a entornos que usan otros algoritmos criptográficos. Un usuario típico de la GSS-API puede ser un protocolo de comunicaciones que usa las funciones de la API para proteger sus mensajes. Un proceso que llama a su GSS-API local acepta tokens procedentes de ésta que transmite a su par remoto. La entidad remota traspasa los tokens recibidos a su implementación de la GSS-API para que los procese. Previamente a la invocación de los servicios de seguridad de la API, las dos aplicaciones que se comunican deben establecer un contexto de seguridad conjunto que contiene información de estado compartida (claves, números de secuencia de mensajes,...). Al concluir la sesión de comunicaciones cada aplicación borra el contexto de seguridad. Como se dijo anteriormente la especificación GSS-API es inde-

pendiente del lenguaje de programación y es necesario especificar estructuras de datos específicas del lenguaje para ser usadas en los parámetros de las funciones GSS-API. La especificación RFC2744 define las estructuras de datos usadas en C y la especificación RFC5653 define las estructuras de datos para Java.

La API PKCS #11 se denomina *Cryptoki*. Es de más bajo nivel que GSS-API y permite un control más directo sobre el uso de las funciones criptográficas. No es independiente del lenguaje ya que usa ANSI C y tampoco es independiente del algoritmo ya que el conjunto de funciones que proporciona permite especificar exactamente cómo se deben llevar a cabo las operaciones criptográficas. A diferencia de la GSS-API fue diseñada desde su inicio como una interfase para HSMs (smart cards, tarjetas PCMCIA,...). A pesar de que PKCS #11 sea una librería nativa en C, existen librerías Java que envuelven PKCS #11 y la hacen accesible desde programas en Java.

En los últimos años se ha detectado un cierto número de errores en CAPIs ampliamente extendidas y esto ha obligado a plantear muy cuidadosamente el diseño de CAPIs.

3.9. Interoperabilidad

Al hablar de interoperabilidad se pueden identificar tres áreas:

- Interoperabilidad a nivel de componentes.
- Interoperabilidad a nivel de aplicaciones.
- Interoperabilidad a nivel de dominios.

La interoperabilidad a nivel de componentes se ocupa de la relación entre sistemas que soportan o consumen servicios PKI (CAs, RAs, entidades finales, repositorios,...) e incluye las siguientes consideraciones:

- Los protocolos, formatos de mensajes y formatos de certificados deben ser comunes entre los componentes PKI.
- Los algoritmos de autenticación de entidad y protección de los datos intercambiados entre componentes PKI deben ser comunes.

- Debe proveerse un procedimiento para facilitar el almacenamiento y recuperación de certificados y su información de estado entre el repositorio y los restantes componentes PKI.
- Las claves privadas deben ser accedidas por los usuarios finales autorizados de forma segura independientemente del método de almacenamiento (software, token hardware, smartcard,...).
- Debe soportarse uno o más mecanismos de estado de los certificados.

La noción tradicional de interoperabilidad a nivel de aplicación se ocupa de la compatibilidad entre pares independientemente del suministrador de la aplicación. Por ejemplo, dos clientes de e-mail S/MIME deben ser capaces de comunicarse entre sí incluso si el software está suministrado por fabricantes diferentes, las plataformas son distintas y cada cliente S/MIME usa tecnología PKI de un vendedor diferente.

Además de las consideraciones relativas a la interoperabilidad a nivel de componentes, la interoperabildad a nivel de aplicaciones incluye las siguientes consideraciones:

- Los certificados y la información de estado de los certificados debe ser compatible.
- Los certificados deben usarse consistentemente con su uso previsto y cualquier restricción asociada.
- Los algoritmos, incluidos los algoritmos criptográficos y el tamaño de las claves, deben ser compatibles.
- Los formatos de codificación de ficheros y mensajes deben ser compatibles.
- Los protocolos de comunicación subyacentes entre pares deben ser compatibles.
- Cualquier método para compartir información relativa a la PKI debe ser compatible.

Otro aspecto de la interoperabilidad a nivel de aplicación involucra el soporte de múltiples aplicaciones de diferentes fabricantes sobre el mismo sistema final. Esto implica acceso, a veces simultáneo, a las credenciales PKI (claves privadas y certificados).

La interoperabilidad a nivel de dominios es probablemente la más complicada de las tres áreas ya que involucra, entre otras cosas, la cooperación de múltiples dominios administrativos. Este tipo de interoperabilidad incluye cuestiones tecnológicas y políticas. Todas las consideraciones necesarias para facilitar la interoperabilidad a nivel de aplicaciones también son aplicables ya que la interoperabilidad a nivel de dominios pretende comunicar aplicaciones en diferentes dominios. Además deben tenerse en cuenta las siguientes consideraciones:

- Se requiere un método para establecer relaciones de confianza entre dominios.
- Información PKI de un dominio debe hacerse accesible al otro y viceversa.
- Cada dominio PKI debe adherirse a ciertas políticas y hacerlas cumplir.

3.10. Interoperabilidad entre dominios

Si bien los estándares cumplen un papel importante en el establecimiento de implementaciones compatibles, la experiencia demuestra que son insuficientes para garantizar la interoperabilidad entre diversos vendedores. Este fenómeno no es exclusivo de la industria de las PKIs, pero es algo de lo que los fabricantes de PKI se deben ocupar. La interconexión de múltiples dominios PKI basados en tecnología suministrada por diferentes fabricantes provoca la aparición de casi todas las facetas concebibles relacionadas con la interoperabilidad. Hay tres áreas que deben considerarse al estudiar las diferentes alternativas. En primer lugar las consideraciones técnicas como protocolos, estructuras de datos y otros aspectos necesarios para facilitar la interoperabilidad. Esta es la área mejor comprendida. En segundo lugar las consideraciones de tipo político. En última instancia las organizaciones necesitan intercambiar información entre ellas basadas en una o más aplicaciones. Los acuerdos que alcancen las organizaciones deben hacerse cumplir a nivel técnico. En último lugar se hallan las consideraciones legales que son, tal vez, las peor comprendidas. Entre ellas se encuentran las cuestiones relacionadas con las firmas digitales y las cuestiones relacionadas con las responsabilidades y obligaciones de las partes. Esta sección se ocupa principalmente de las consideraciones técnicas y políticas.

Junto con el desarrollo del estándar se deben realizar dos tareas que ayudan a alcanzar el nivel deseado de interoperabilidad multivendedor.

En primer lugar se deben adaptar los estándares a un entorno particular. El propósito de estos perfiles es identificar las características genéricas del estándar que son obligatorias, opcionales o prohibidas. Suelen incluir también reglas de uso y guías de implementación. En el caso específico de las PKIs es necesario adaptar protocolos PKI, el esquema y los protocolos asociados con el repositorio auxiliar y los certificados y CRLs. En particular los certificados y CRLs incluyen extensiones que deben especificarse para eliminar las ambigüedades y determinar sus usos particulares en el perfil. Por ejemplo el RFC5280 *Internet X.509 Public Key Infrastructure Certificate and CRL Profile* define un perfil del estándar X.509 para su uso en Internet.

En segundo lugar, para implementar la interoperabilidad multivendedor es necesario establecer escenarios de interoperabilidad y realizar tests sobre esos escenarios. Una posibilidad consiste en construir centros de tests de interoperabilidad neutrales respecto al fabricante que evalúen la conformidad de un producto dado contra un conjunto específico de criterios. Complementariamente se puede participar en iniciativas de interoperabilidad patrocinadas por organizaciones gubernamentales o por algún sector de la industria. A continuación se detallan algunas de estas iniciativas.

El *PKI Interoperability Expert Group* que es parte del Foro de Cooperación Económica Asia-Pacífico (APEC) ha examinado la interoperabilidad entre los diferentes esquemas de los países que lo forman. El proyecto pretende identificar los elementos claves de un esquema PKI y mapear las diferentes aproximaciones en esos elementos clave. El proyecto reconoce la existencia de múltiples niveles de certificados y se centra en desarrollar un certificado que tendrá efectos legales en todas las economías APEC independientemente de su lugar de emisión. Los esquemas bajo consideración son gubernamentales o basados en estándares como los desarrollados por el *Certification Forum of Australia*. La existencia de estos esquemas de acreditación facilitará el concepto de *reconocimiento cruzado* previamente establecido por el *PKI Interoperability Expert Group*. Este grupo trabaja en estrecha colaboración con el *European Electronic Signature Standards Initiative* (EESSI) para garantizar que no hay inconsistencias entre ambos grupos. El grupo ha detectado carencias en los estándares de seguridad de las propias CAs y ha discutido la necesidad de pruebas piloto de interoperabilidad entre las distintas economías que involucran cuestiones legales, técnicas y políticas.

El *Communications-Electronics Security Group* (CESG) dependiente del gobierno británico ha realizado pruebas de interoperabilidad bajo los auspicios del proyecto *CLOUD COVER*. El proyecto pretende establecer estándares para promocionar el desarrollo por parte de la industria de productos y servicios PKI que se ajusten a los requisitos de distribución de claves del propio gobierno británico. Este proyecto ha detectado un buen número de problemas relacionados con la interoperabilidad entre productos de diferentes fabricantes:

- Codificación y decodificación de la información (diferencias en el uso de la codificación BER/DER para las extensiones, uso de OIDs no estándar, formatos de fecha inconsistentes,...).
- Problemas relacionados con límites o rangos (número de serie de certificados, longitud de los nombres, ...).
- Convenciones relativas a los nombres (orden de los atributos de los DN,...).
- Cuestiones relacionadas con certificados, CRLs y caminos de certificados (restricciones en la longitud de los caminos de certificados, uso inconsistente de algunos campos en certificados y CRLs,...).
- Cuestiones relacionadas con directorios.

El *Government of Canada Public Key Infrastructure* (GoC PKI) define una metodología detallada de cross-certificación. Describe un proceso paso a paso que incluye formularios y requisitos específicos para la documentación de CPs y CPSs. El proceso en cuestión sigue las fases para concertar un contrato entre las partes en cuestión.

El *US Federal Bridge CA Project* pretende proporcionar interoperación entre PKIs de departamentos y agencias federales. Posteriormente este proyecto pretende proporcionar mecanismos para la interoperabilidad entre las administraciones federales y la estatal, entre diferentes estados y entre PKIs del sector comercial. La interoperabilidad entre gobiernos federales presenta dificultades técnicas y organizativas. Por un lado los diferentes departamentos y agencias usan productos de diferentes fabricantes que no fueron diseñados para interoperar. Por otro lado usan diferentes CPs. Además, intentar imponer una jerarquía en una estructura tan grande como la de los gobiernos federales

es muy complejo y completamente imposible entre PKIs de diferentes estados. El gran número de agencias y departamentos federales hace imposible la cross-certificación bilateral o las listas de confianza. Como resultado de todo esto el *Federal Public Key Infrastructure Technical Working Group* bajo la dirección del *US National Institute of Standards and Technology* (NIST) adoptó el concepto de *Bridge CA* (CA puente) del sector comercial que en aquel entonces estaba afrontando un problema similar al intentar integrar un elevado número de PKIs no jerárquicas. El concepto de *Bridge CA* permite reducir la carga de trabajo asociada con la gestión y administración de los cross-certificados. La idea de la *Bridge CA* consiste en el despliegue de un *puente de confianza* entre las diferentes PKIs mediante la cross-certificación con *Principal Certification Authorities* en los distintos departamentos y agencias federales.

La estructura en CA puente difiere de la jerárquica en que su clave pública no constituye el punto de confianza. Esto es importante por varias razones. En primer lugar las claves de los puntos de confianza deben ser suministradas por mecanismos seguros fuera de banda. Por lo que cambiar el punto de confianza es muy difícil para una organización grande y heterogénea. Además, si la CA puente fuese la raíz de la jerarquía, las PKIs federales cederían una buena parte del control de sus PKIs. A causa de los diferentes requisitos de las CPs de las diferentes agencias la estructura jerárquica se tornaría excesivamente compleja. La CA puente permite que las agencias y departamentos federales mantengan el control de su PKI local y de su CP. Mediante la elección cuidadosa de los certificados que emite a la CA puente se puede regular el grado de confianza que concede a los certificados aceptados a través de la CA puente.

Las PKIs de los departamentos federales y agencias se desarrollaron independientemente con diferentes requisitos de seguridad e implementando un amplio rango de CPs. Una cadena de certificados entre una PKI con unos requisitos bajos de seguridad y una PKI con unos requisitos altos reduciría la seguridad de los usuarios de la PKI con requisitos altos si no se tomasen medida para evitarlo. La CA puente federal usa *policy mapping* del estándar X.509 para prevenir estos problemas. El *policy mapping* permite a una CA afirmar en un cross-certificado que las CPs de una PKI cross-certificada son equivalentes a las del dominio local. El *Federal Policy Authority* establece la política de la CA puente que sirve como base para la emisión de certificados desde la CA puente. Actualmente, la política de la CA puente establece cuatro niveles de seguridad: rudimentario, básico, medio y alto. La *Federal Policy Authority*

evalúa las CPs de las PKIs candidatas y mapea las políticas de las agencias con las política de la CA puente federal; análogamente, las diferentes agencias evalúan la políticas de la CA puente y la mapean con las propias. Las cadenas resultantes de certificados reflejan estas decisiones políticas y permiten a las aplicaciones de las partes confiantes aceptar o rechazar los certificados. Por supuesto, la información de políticas incluida en los certificados es útil sólo si las aplicaciones son capaces de procesarla.

El uso de la CA puente impone requisitos especiales en las aplicaciones clientes. Por un lado la CA puente combina arquitecturas PKI jerárquicas y distribuidas en una única PKI en malla. Las aplicaciones que construyen cadenas de certificados de PKIs jerárquicas deben ser mejoradas para desarrollar cadenas de certificados de PKIs en malla. Por otro lado, además de las extensiones *policy mapping* pueden usarse otras extensiones para prevenir la reducción de seguridad entre las PKIs como por ejemplo *nameconstraints*, esto obliga a las aplicaciones a tratar un gran número de extensiones.

Los directorios (repositorios) es la forma más común de compartir certificados y su información de revocación. Las PKIs de las agencias y departamentos federales se desarrollaron sin considerar aspectos relativos a la compatibilidad e interoperabilidad. Análogamente los directorios se desarrollaron sin pensar en una posterior integración en entornos de varios fabricantes. Para que el concepto de CA puente funcione en el gobierno federal de los Estados Unidos la cuestión de la interoperación entre directorios de diferentes fabricantes debe resolverse.

Los protocolos propietarios y las limitaciones en los clientes son barreras técnicas para la interoperación de directorios, pero también hay problemas políticos. Los directorios contienen a menudo información sobre los empleados de la empresa que no puede ser revelada a las otras organizaciones conectadas a la CA puente. Tanto las barreras técnicas como políticas pueden abordarse mediante los directorios frontera. El concepto de directorios frontera fue desarrollado para proporcionar una parte de la información del directorio a los aliados, mientras otra información permanece exclusivamente en el directorio local. La idea consiste en ubicar un directorio fuera del *firewall* de la empresa. Un subconjunto de la información del directorio se exporta del directorio interno al directorio frontera y se hace así disponible al conjunto de PKIs. La CA puente federal debe proporcionar su propio servicio de directorio para sus certificados y CRLs junto con un servicio de directorio frontera para aquellas agencias y departamentos federales pequeños que no pueden desplegar sus

propios sistemas de directorio frontera.

La interoperabilidad de directorios y CAs junto con las limitaciones de los clientes son dificultades que la tecnología actual ya ha superado.

3.11. Funciones off-line

Una cuestión que debe considerarse durante el diseño de una PKI es la funcionalidad de seguridad disponible para un usuario desconectado de la red. Es evidente que algunos servicios como el no-repudio, recuperación de claves, renovación de claves, acceso a certificados o información de revocación,... no pueden ser soportadas off-line. Dependiendo de las políticas del entorno la incapacidad de acceder a tales servicios puede tener implicaciones en la seguridad y limitar las tareas PKI que el usuario puede realizar off-line. La política de la empresa dictará explícitamente si la funcionalidad off-line es valiosa para el entorno. Esta decisión dependerá en gran medida de los tipos de aplicaciones empleadas por los usuarios y de si las aplicaciones requieren soporte PKI on-line y off-line.

3.12. Seguridad física

Los componentes PKI más sensibles deben protegerse en entornos de alta seguridad para impedir su modificación, destrucción o accesos no deseados. La seguridad física incluye:

- Restringir o eliminar el acceso a través de la red.
- Ubicar los componentes sensibles en habitaciones con control de acceso.
- Proteger apropiadamente las copias de seguridad en cintas u otros soportes.

Hay que buscar un compromiso entre las medidas de seguridad del sistema y su facilidad de uso. Una seguridad muy restrictiva introducirá más retrasos y complejidad operativa que un sistema con una seguridad más relajada.

La seguridad física que debe incorporarse en una instalación PKI debe decidirse antes del despliegue. Esta decisión dependerá de los requisitos de

seguridad del sistema, una completa evaluación de riesgos de la operación prevista y la degradación tolerable en la facilidad de uso del sistema. En muchos entornos una CA off-line, es decir una CA sin conexión a la red y ubicada en una habitación con control de acceso, junto con una RA on-line pueden ofrecer un compromiso razonable.

3.13. Componentes hardware

Una instalación PKI puramente software es perfectamente viable en algunos entornos. No obstante, los sistemas operativos y las aplicaciones son vulnerables a virus, hackers, troyanos e incluso a usuarios que sin ninguna malicia pueden originar una brecha en la seguridad. Por todo ello y para ganar seguridad adicional se suelen usar algunos componentes hardware:

- Dispositivos hardware para realizar operaciones criptográficas.
- Smartcards.
- Dispositivos biométricos que habilitan la identificación de usuario multifactor.

Del mismo modo que con la seguridad física hay un compromiso entre los componentes hardware y la facilidad de uso del sistema. Con la adición de componentes hardware la degradación en la simplicidad de uso puede ser muy grande y afectar a todos los usuarios. La adición de componentes hardware es una consideración operativa significativa porque puede tener efectos adversos en el rendimiento (limitaciones en las smartcards por ejemplo), en la aceptación por parte de los usuarios y en los costes totales de despliegue.

3.14. Consideraciones sobre el despliegue

Puesto que implantar una PKI es una tarea compleja, la mayoría de organizaciones escogen desplegar la PKI en fases. En primer lugar se selecciona una aplicación PKI piloto. Las fases que habitualmente se suelen seguir son las siguientes:

- En primer lugar hay que evaluar el software de los diferentes vendedores. Se debe tener en cuenta el número de usuarios que soporta, la

facilidad de administración, tipos de certificados que soporta,... Al finalizar esta fase se debe haber escogido un fabricante principal y uno de respaldo.

- Seguidamente se debe negociar el contrato final con el vendedor. Si las necesidades de la organización no coinciden con las prestaciones del producto puede ser necesario negociar adaptaciones del software. Esto suele conllevar un incremento sustancial del coste.

- El siguiente paso consiste en acondicionar una instalación segura para el servidor CA e instalarlo, configurarlo y realizar pruebas sobre él. También se debe hacer lo mismo con la RA. Se deben llevar a cabo pruebas de integración con el directorio.

- A continuación se efectúa una fase piloto que permite ganar experiencia operativa con el producto y la aplicación PKI antes del despliegue completo. Se emite un número de certificados limitado a usuarios que también instalarán la aplicación piloto PKI. En esta fase se deben emitir certificados de todos los tipos. La respuesta de los usuarios a esta primera aplicación es un claro indicador de las dificultades que surgirán en el despliegue completo. Puesto que durante esta fase surgen problemas, los certificados emitidos durante esta fase suelen tener un periodo de vida breve.

- A continuación se realiza un despliegue limitado. Probablemente se reinstalará el software en los servidores. El personal de soporte recibirá formación que incluirá los problemas detectados en la fase piloto. Los usuarios también serán adecuadamente formados en el uso de la PKI.

- Finalmente se realiza el despliegue completo. Se instala la aplicación PKI y se emiten certificados para todos los usuarios. El número de fases de esta etapa dependerá del número de certificados a emitir, de la distribución geográfica de los usuarios, de la capacidad del personal para soportar una población creciente de certificados y de la velocidad con la que son formados los usuarios y el personal de soporte.

Para cada fase del despliegue se debe determinar su duración y los recursos dedicados. Se debe decidir la ubicación de los servidores de producción y proporcionar pautas para su operación. En cuanto a la formación, se debe

decidir si será interna o externa y actuar en consecuencia. Se debe determinar quién tendrá la responsabilidad de seleccionar el producto, negociar la compra y proporcionar el hardware. Finalmente también es crucial decidir la procedencia de los fondos del presupuesto.

Capítulo 4

Mantenimiento

4.1. Introducción

Este capítulo se centra en las tareas relacionadas con el mantenimiento y la operación diaria de una PKI y está dividido en tres partes. En primer lugar se analizan las diversas fases del ciclo de vida de certificados y claves. Seguidamente se estudian los protocolos que soportan las distintas fases del ciclo de vida. Finalmente se realizan una serie de consideraciones sobre distintas tareas que se deben llevar a cabo durante el mantenimiento de una PKI y las personas que las llevan a cabo.

4.2. Ciclo de vida de certificados y claves

La gestión de un sistema de clave pública es un problema complejo, aunque más sencillo que la gestión de un sistema puro de claves simétricas. Una PKI realiza múltiples funciones relacionadas con la creación, emisión y subsiguiente cancelación de las claves pública y privada y sus certificados asociados. Estas funciones constituyen el día a día de la operación de las PKIs y se engloban bajo la denominación de *ciclo de vida de certificados y claves*.

En esta sección se van a comentar las distintas funciones del ciclo de vida que una PKI exhaustiva debe ofrecer y que son las siguientes: inicialización de entidades finales, generación de claves, almacenamiento de claves privadas, registro de entidad final, prueba de posesión, generación de certificados, distribución de certificados y claves, diseminación de certificados, archivado y recuperación de claves, actualización de claves de entidad final, actualización

de claves de CAs, recuperación de certificados y CRLs, construcción y validación de caminos de certificación, revocación de certificados y expiración de certificados.

Es importante destacar que las funciones del ciclo de vida deben automatizarse tanto como sea posible y evitar que el usuario final deba intervenir. El estándar ISO/IEC 11770-1 se ocupa de especificar un modelo general para el ciclo de vida proporcionando definiciones y los servicios necesarios, aunque no detalla los mecanismos usados.

4.2.1. Inicialización de entidades finales

Antes de que una entidad final pueda hacer uso de la PKI debe inicializarse con algunas informaciones. En la práctica la información suele proporcionarse cuando se configura el sistema o a través de la red de comunicaciones.

La entidad final debe ser informada de los servicios que ofrecen los diferentes componentes de la PKI y dónde se debe conectar para obtener los mencionados servicios. Por ejemplo, se deben configurar los nombres de red o direcciones IP de las RAs, CAs o cualquier otro servicio de la PKI.

Durante esta fase también se transmiten a la entidad final la lista de los certificados de confianza como, por ejemplo, sería el caso de los certificados configurados en el navegador.

Otra información que suele proporcionarse en esta fase es el nombre de la propia entidad final u otra información necesaria para establecer su identidad en la solicitud del certificado.

4.2.2. Generación de claves

La generación de claves consiste en la generación del par de claves pública y privada. Son diversos los aspectos relativos a la generación de claves que merecen una especial consideración.

Un principio importante a tener en cuenta es la separación de claves. En pocas palabras la separación de claves consiste en que cada par de claves debe usarse sólamente para un propósito único. Si las mismas claves se usan para diferentes propósitos es muy difícil determinar la seguridad global del sistema. Esto implica que cada aplicación use su propio conjunto de claves. La mayoría de aplicaciones PKI usan un único par de claves, pero las hay que usan dos pares e incluso hasta tres pares. La mayoría de aplicaciones de

correo electrónico seguro usan dos pares de claves. Uno de ellos permite ofrecer el servicio de recuperación de claves de cifrado, mientras que el otro par soporta firmas digitales y no-repudio. Los servicios de no-repudio y cifrado tienen diferentes requisitos. Por un lado, el no-repudio y las firmas digitales requieren que el acceso y uso de la clave privada esté estrictamente limitado a un usuario. Por otro lado, la recuperación de claves precisa que la clave privada se archive en alguna base de datos centralizada. En una aplicación que usa tres pares de claves, un par se usa para identificar al usuario, otro par se usa para firmas digitales y el tercer par para cifrado.

El servicio de no-repudio tiene unas características que lo diferencian de los demás servicios. Para poder atribuir una firma a un individuo es necesario garantizar que la clave privada sólo puede ser accedida por el usuario en cuestión. Esto tiene algunas implicaciones sobre la clave privada: no pueden existir copias de la clave, la clave debe almacenarse en una ubicación segura apropiada y debe autenticarse el acceso a la clave antes de proceder a una operación de firma. Preferiblemente, las claves deben generarse localmente en algún dispositivo que disponga de almacenamiento protegido como por ejemplo una smart card. Además no conviene archivar la clave privada porque puede quedar expuesta a operadores poco éticos, transportes inseguros u otros posibles peligros.

Una controversia relativa a la generación de claves es qué componente de la PKI se encarga de ella. El estándar X.509 no determina qué parte de la PKI se ocupa de la generación de claves. Puede ser el sistema del usuario final, una CA, una RA o incluso una tercera parte fiable. Son muchos los criterios que influyen en la elección de la ubicación de la generación de claves. Aunque no hay unanimidad sobre el tema, suele aceptarse que la generación de claves debe tener lugar en el sistema del usuario final si las claves se van a usar para el propósito de no-repudio. Por otro lado, una generación de claves centralizada facilita servicios como el archivado, permite la generación y distribución de claves por lotes y, además, es susceptible de usar smart cards. No obstante, el componente encargado de la generación centralizada de claves constituye un blanco para un ataque y los empleados que tienen acceso al sistema deben ser de total confianza. También hay que considerar que la generación de claves requiere abundantes recursos de CPU y dispositivos muy restringidos, como es el caso de teléfonos móviles, no la pueden llevar a cabo. En tales entornos puede tener sentido delegar la generación en sistemas más potentes, como una CA. Sin embargo, esta solución no escala bien. También las políticas de la or-

ganización influyen. Puede darse el caso que la organización requiera que las claves se generen en un módulo criptográfico altamente confiable y evaluado independientemente. Tales módulos son muy caros y por ello en esos casos las claves se generan centralizadamente. En otras ocasiones las políticas requieren el uso de smart cards para almacenar las claves privadas. En tal caso, para claves de firmas digitales que no requieren ser archivadas esto funciona bien, pero si las claves de cifrado son generadas en la smart card no hay manera de copiarlas en un archivo. Finalmente, simples consideraciones legales obligan en ocasiones a delegar la responsabilidad de la generación de claves en algún componente más fiable que una entidad final.

En resumen, son muchos los factores que influyen en la elección de la ubicación de la generación de claves. En general, las claves se pueden generar en cualquier componente (CA, RA, entidad final) y deben ser los protocolos los que permitan transmitir las claves de forma segura entre estos sistemas. Esto está explícitamente indicado en las especificaciones RFC4210 y X.509. Muchas implementaciones PKI usan un modelo dividido que genera las claves de cifrado centralmente y las claves de firma digital en la aplicación cliente o en una smart card.

4.2.3. Almacenamiento de claves privadas

Esta sección se ocupa de las técnicas usadas por las PKIs para almacenar las claves privadas de los usuarios finales, no las claves privadas de las CAs u otras entidades.

Un concepto fundamental de las PKIs es que la clave privada debe estar únicamente en posesión del usuario cuya identidad está declarada en el certificado digital correspondiente. Garantizar la seguridad y singularidad de la clave privada requiere un esfuerzo considerable. Aspectos del uso de las claves privadas que son críticos son la ubicación de su almacenamiento, la protección de la clave privada y la manera en que un usuario valida su identidad para tener acceso a la clave.

El compromiso de una clave privada puede conllevar inconvenientes significativos y consecuencias difíciles de tratar. Las consecuencias específicas del compromiso de una clave privada de una entidad final dependen del tipo de la clave. Si, por ejemplo, se compromete una clave privada de cifrado o de intercambio de claves, el usuario en cuestión no sólo debe revocar inmediatamente el certificado correspondiente sino que también debe localizar

todos los documentos cifrados con una clave simétrica protegida con la clave privada comprometida. Estos documentos deben ser de nuevo protegidos. El problema es que pueden existir copias dispersas de estos documentos muy difíciles de localizar. Si, por contra, se compromete una clave privada dedicada a firmar documentos puede bastar con que el usuario final revoque el certificado correspondiente, aunque todos los documentos firmados con dicha clave pueden ser dudosos.

La sensibilidad de la información protegida determinará el tipo de protección que requiere la clave privada. Básicamente, hay dos tipos de técnicas: software y hardware.

Las técnicas software almacenan la clave privada en ficheros cifrados. La forma más simple de mantener una clave privada cifrada es la envoltura proporcionada por el estándar PKCS #8 de RSA. En este caso la clave privada se cifra usando un algoritmo de clave simétrica. PKCS #8 especifica una estructura de datos que además de la clave privada incluye otra información como el OID del algoritmo con el que puede usarse la clave privada. Este estándar puede usarse combinado con el estándar PKCS #5 que explica cómo derivar una clave simétrica partiendo de una password. Para ello se vale de funciones hash como MD2, MD5 o SHA-1. Así pues el usuario escribe una password, de la cual se deriva una clave simétrica, para acceder a su clave privada. Existen diversas técnicas similares a esta que son usadas por las diferentes implementaciones de PKIs.

El estándar PKCS #12 proporciona un método más sofisticado para la protección no sólo de claves privadas, sino para la protección de cualquier objeto en general. Este estándar se usa para intercambiar credenciales PKI entre aplicaciones y adopta normalmente la forma de un fichero. Un usuario puede usar diferentes aplicaciones de e-mail o navegadores y desear ser reconocido siempre igual. Por otro lado, un mismo usuario puede usar diferentes máquinas y en todas ellas usar el mismo conjunto de claves. PKCS #12 proporciona una forma de sincronizar todas estas aplicaciones.

Sin embargo, las técnicas software plantean graves problemas. Los ficheros pueden copiarse fácilmente. Las passwords elegidas pueden ser poco seguras. Cuando una aplicación legítima descifra la clave privada, ésta es copiada en memoria y puesto que la memoria de trabajo puede copiarse en disco existe siempre el riesgo de dejar una clave privada en el disco. Por todos estos motivos y algunos más, una buena implementación de hardware es siempre más segura.

Las smart cards son los dispositivos hardware más ampliamente usados para almacenar claves privadas y otras informaciones criptográficas. Junto con las claves privadas, también pueden almacenar certificados, claves simétricas, passwords,.... Son básicamente pequeños microprocesadores con memoria incrustados en plástico del mismo tamaño que una tarjeta de crédito. Además, disponen de memoria segura y resistente a alteraciones para almacenar información sensible. Muchas smart cards soportan la generación de claves en su interior y también pueden realizar cálculos criptográficos en el procesador que incluyen. Las smart cards aceptan datos del exterior, los procesan con programas almacenados en su interior y emiten el resultado al exterior. Son dispositivos con 2 factores de autenticación: algo que se tiene (la propia smart card) y algo que se sabe (PIN) por lo que ofrecen más seguridad que las técnicas software anteriormente mencionadas.

Las smart cards presentan el inconveniente de ser lentas. Por ello no son capaces de realizar cualquier operación criptográfica. Por ejemplo, si se pretende realizar una firma digital sobre un documento de 1 Mbyte uno de los primeros pasos del proceso es producir un hash del documento. Si se realizase el hash dentro de la smart card tendríamos que enviar 1 Mbyte de texto a través de un interface half duplex a 9600 bps lo cual puede llevar 15 minutos. Por la misma razón no se pueden realizar descifrados simétricos de grandes cantidades de datos. Lo que se hace es realizar el hash en el PC y, a continuación, enviarlo a la smart card donde simplemente es firmado por la clave privada. Análogamente los descifrados simétricos se realizan en el PC, la clave simétrica cifrada es lo único que se transfiere a la smart card para que sea descifrada con la clave privada.

Otra ventaja de las smart cards es que son portables y, como resultado, una persona puede transportar sus claves privadas donde quiera que vaya. Se resuelven de esta manera muchos de los problemas derivados de las copias de ficheros.

Las claves privadas pueden usarse para cifrar o para firmar digitalmente documentos. Como ya se ha dicho, estos dos tipos de claves presentan diferentes requisitos. Cuando se requiere algún tipo de sistema de archivo de claves es importante guardar una copia de la clave privada de cifrado. Por otro lado, no interesa hacer copias de las claves privadas usadas para firmar. Para resolver esta dualidad, las smart cards soportan la posibilidad de generar las claves interna o externamente. El par de claves usado para firmas digitales se genera dentro de la tarjeta. La clave privada nunca abandona la tarjeta,

mientras que la clave pública correspondiente se transmite fuera de la tarjeta para solicitar el certificado. Nadie puede leer la clave privada en el interior de la smart card y todas las operaciones criptográficas que involucren a la clave privada deben ocurrir dentro de la tarjeta. La situación es diferente con una clave de cifrado. En muchos casos es deseable mantener una copia de la clave privada de cifrado para propósitos de recuperación. Si, por cualquier motivo, se pierde la smart card y sólo hay una copia de la clave privada de cifrado sería imposible recuperar los mensajes cifrados. Este no es el caso con la clave privada destinada a firmas digitales, puesto que basta con el certificado, públicamente disponible, para validar la firma. Por todo ello, las claves de cifrado son generadas externamente a la smart card, si dicha clave se generase internamente la smart card no permitiría copiarla. Una vez las claves han sido generadas externamente, la clave privada se archiva y es transferida a la smart card, mientras que la clave pública es certificada.

Las smart cards también presentan inconvenientes cuando se usan en una PKI. El primer problema es que la mayoría de los sistemas no incluyen lectores de smart cards y un problema relacionado es que los lectores requieren drivers. Un segundo inconveniente es el coste de las tarjetas y los lectores. Con el paso del tiempo estos problemas se van resolviendo y, poco a poco, la tecnología de smart cards se va extendiendo. Los sistemas PKI se beneficiarán ampliamente de la adopción de esta tecnología.

4.2.4. Registro de entidad final

El registro de una entidad final es el proceso mediante el cual la identidad de un usuario individual, aplicación o dispositivo hardware es establecida. El proceso de registro, al igual que la mayoría de aspectos de la PKI, está regulado por los documentos *Certificate Policies* (CP) y *Certificate Practice Statements* (CPS). Concretamente estos documentos determinan el nivel de verificación asociado con la evaluación de la identidad de una entidad final.

Un posible mecanismo es que los usuarios se presenten en persona y proporcionen evidencias de su identidad (DNI, permiso de conducir, pasaporte,...) a un operador de registro. El operador de registro verificará los datos y creará una solicitud de certificado con su propia firma electrónica. La CA recibirá la solicitud y emitirá el certificado. El certificado puede ser entregado al usuario final en una smart card. Este mecanismo de verificación de la entidad es seguro, pero costoso.

Una alternativa para organizaciones que no requieran una seguridad tan elevada consiste en proporcionar al usuario final un identificador junto con una password a través de un mecanismo fuera-de-banda como un e-mail o una llamada telefónica. Cuando el propio usuario se registra on-line presenta el identificador y la password lo que posibilita su identificación por parte del sistema. La información necesaria para la certificación la puede proveer el propio usuario o bien una base de datos de personal adecuadamente conectada.

Otro método posible es similar al que se usa en la expedición de tarjetas de crédito. Una central emite smart cards personalizadas a partir de información que puede ser recopilada desde bases de datos de personal. Con la información obtenida de las bases de datos se construye una solicitud de certificado que se envía a la CA para su certificación. La validación de la identidad del usuario final se hace enviando la smart card a su dirección postal y comunicando el PIN mediante una llamada telefónica.

4.2.5. Prueba de posesión (POP, Proof of Possession)

Según el estándar RFC4210 parte del proceso de registro consiste en verificar que la entidad final dispone de la clave privada correspondiente a la clave pública que se registra. El nombre de esta verificación es POP (Proof of Possession). En la actualidad hay muchos protocolos operativos (entre ellos varias aplicaciones de e-mail) que no comprueban la asociación entre la entidad final y la clave privada. Mientras las aplicaciones no realicen esta verificación no hay otra opción que realizarla durante el proceso de registro.

La POP puede obtenerse de diferentes modos dependiendo del tipo de clave para el que se solicita el certificado. Si la clave va a usarse para firmas digitales, basta con que se firme la solicitud de certificado con la clave privada. La firma puede ser validada por la CA/RA con la clave pública de la misma solicitud. De este modo la clave privada permanece en el almacenamiento seguro de la entidad final y no es necesario que se envíe a la RA/CA. Para claves de cifrado, la entidad final puede proporcionar la clave privada a la RA/CA o bien descifrar un valor con la clave privada. El método de la descifrado puede ser directo o indirecto. El método directo consiste en que la RA/CA emita un reto aleatorio cifrado y la entidad final lo devuelva inmediatamente descifrado. El método indirecto consiste en emitir el certificado cifrado a la entidad final. Esto puede ser aceptable si la PKI no tiene que publicar el certificado en

un directorio. Si las claves se usan para un protocolo de intercambio de claves la única manera de verificar que se tiene la clave privada es realizar un intercambio de claves. Si el par de claves puede usarse para diferentes propósitos incluido el de firma es razonable apoyarse en el mecanismo de firma como prueba de existencia de la clave.

4.2.6. Generación del certificado

Independientemente de dónde se haya generado el par de claves, la responsabilidad de la creación del certificado es exclusivamente de la CA. Si la clave pública fue generada por una entidad diferente de la CA, la clave pública debe ser transmitida de forma segura a la CA.

Durante la generación del certificado la CA firmará el contenido del certificado usando su clave privada. El contenido de un certificado depende del contenido de la solicitud del certificado.

4.2.7. Distribución de certificados y claves

Una vez se ha generado el par de claves y el certificado se debe proceder a su distribución. Los requisitos de la distribución dependen de dónde se ha generado el par de claves, el uso previsto del certificado y otras restricciones operativas o de políticas. Si la clave privada no se generó en el sistema cliente debe hacerse llegar al propietario de la clave de forma segura. Un certificado puede distribuirse directamente al propietario, a un directorio o a ambos. Tanto la solicitud del certificado, como el propio certificado y la clave privada deben ser transmitidos a través de protocolos con mecanismos seguros (CMP, CRMF,...). En ocasiones el certificado es almacenado en una Web de la propia CA, desde donde puede ser recuperado por el usuario.

4.2.8. Diseminación de certificados

Para que un certificado sea útil debe hacerse llegar a entidades finales para que, a su vez, puedan cifrar información destinada a otras entidades o bien verificar la firma electrónica de otras entidades. La diseminación de certificados y otras informaciones como CRLs puede llevarse a cabo de diferentes modos.

Una forma básica de diseminación consiste en que los propios usuarios se transmitan entre ellos los certificados a través, por ejemplo, de e-mails.

Este mecanismo no es práctico por distintos motivos: no escala bien, no hay garantía de que la información de revocación se transmita de forma fiable a todos los individuos,... Sin embargo, es un mecanismo que puede funcionar relativamente bien en pequeñas comunidades de usuarios donde se conocen directamente entre ellos o a través de una tercera persona. Un ejemplo típico de este sistema es *Pretty Good Privacy* PGP.

El método más extendido para la diseminación de certificados y otras informaciones relacionadas es la publicación en repositorios capaces de almacenar y diseminar la información. Algunos ejemplos de tales repositorios son:

- Servidores LDAP.
- Servidores X.500.
- Servidores DNS de acuerdo con la especificación RFC4398, Storing Certificates in the Domain Name System.
- Servidores Web de acuerdo con la especificación RFC2585.
- Servidores FTP de acuerdo con la especificación RFC2585.
- Bases de datos corporativas.

Estos repositorios permiten recuperar información de certificados libremente, sin control de acceso de lectura. Sin embargo, se debe controlar el acceso para publicar la información en estos repositorios ya que de lo contrario se podrían cambiar libremente certificados o CRLs. La ubicación de los repositorios puede comunicarse al cliente a través de direcciones IP o nombres DNS. Algunas extensiones de los certificados también se usan para información de ubicación.

Una de las ventajas que presentan los repositorios es que muchas organizaciones ya tienen uno de ellos desplegado y es relativamente simple agregar la información de la PKI. Además, los repositorios permiten establecer relaciones entre personas que no se conocen previamente, a diferencia de PGP. Finalmente, debido a que los certificados y CRLs son estructuras de datos autoprotegidas el repositorio no necesita ser seguro desde la perspectiva de integridad de los datos. Un repositorio también presenta desventajas. La red de comunicaciones tiene que soportar la publicación y recuperación de información relacionada con la PKI. Si la población de la PKI es elevada los

repositorios deben ser capaces de soportar una carga de trabajo elevada. También hay cuestiones relacionadas con la replicación de la información a través de múltiples repositorios (impacto en el rendimiento, retrasos, sincronizaciones,...). Por último, los repositorios están sujetos a ciertos ataques como la denegación de servicio.

También hay que considerar el despliegue de los certificados y CRLs a otros dominios PKIs de otras empresas u organizaciones. En primer lugar, el concepto de un repositorio públicamente accesible entra en contradicción con la consideración de que algunas de las informaciones del repositorio se consideran sensibles y por tanto inapropiadas para una libre distribución. Esto obliga a tomar ciertas medidas en torno a la información que se distribuye. En segundo lugar, hay que considerar la configuración asociada con el despliegue de repositorios entre dominios. Una primera posibilidad es el acceso directo de entidades externas al repositorio corporativo a través del firewall. Puede ser útil cuando la relación de confianza entre los dos dominios es recíproca o cuando el mismo repositorio está protegido contra accesos no autorizados. Esta posibilidad requiere también de un mecanismo de confidencialidad para transmitir la información (TLS o IPSec). Una segunda opción es replicar la información relevante del repositorio a otro repositorio externo al firewall o bien usar un servidor proxy externo que reorienta las peticiones externas hacia el repositorio interno. Una tercera opción consiste en usar un repositorio compartido en el que cada PKI deposita la información pertinente relativa a certificados y CRLs de modo que los otros dominios PKI pueden recuperar la información cuando la necesiten. Por último, pueden usarse mecanismos de replicación interdominios que copian la información de certificados y CRLs aplicable directamente de un dominio a otro y viceversa. La posibilidad de automatizar la replicación interdominios depende de los protocolos usados. Por ejemplo, si ambos dominios soportan Servicios de Directorio basados en X.500 pueden usar Directory Information Shadowing Protocol (DISP).

La diseminación de certificados también puede ser soportada en el intercambio del protocolo de comunicaciones. Tal es el caso del correo electrónico basado en S/MIME versión 3, TLS o IPSec. En algunos entornos este puede ser el único mecanismo viable. Por ejemplo, Internet usa este mecanismo porque carece de un repositorio global para soportar la diseminación de certificados y CRLs. Estos protocolos pueden complementar el repositorio en lugar de reemplazarlo. Por ejemplo, el certificado de verificación del originador del mensaje se puede enviar junto con el e-mail firmado. Esto permite al receptor

verificar la firma digital sin necesidad de recuperar el certificado de un repositorio. No obstante, el certificado de cifrado del destinatario del correo sí debe ser obtenido de un repositorio en origen para poder cifrar el mensaje.

A medida que los entornos PKI escalan a decenas de miles o incluso millones de usuarios la diseminación puntual de los certificados y CRLs es una cuestión crítica. Este es uno de los requisitos fundamentales para el despliegue exitoso de una PKI a gran escala.

4.2.9. Archivado y recuperación de claves

Una PKI completa debe ofrecer un servicio de archivado y recuperación de claves. Los motivos para la implantación de esta función se deben a la posible pérdida de las claves privadas de cifrado. Un usuario puede olvidar la password de cifrado de un fichero de claves, dejar la compañía o morir inesperadamente en un accidente; una smart card puede perderse, dañarse o robarse; un disco puede corromperse,... En estas circunstancias es necesario un proceso de recuperación de las claves privadas de cifrado, de lo contrario es imposible recuperar la información cifrada. La incapacidad de recuperar estos datos puede tener severas consecuencias operativas o financieras, particularmente en entornos corporativos, por lo que cualquier compañía u organización establece en sus documentos de políticas procedimientos para recuperar dicha información. Por todo lo dicho anteriormente, la necesidad de un sistema de archivado y recuperación de claves se basa en principios sólidos y prácticos.

Las claves destinadas a firmas digitales no deben ser archivadas para garantizar el no-repudio. Si existen copias de la clave privada para firmas digitales, el propietario de la clave puede argumentar que otra persona proporcionó su firma a un documento conflictivo. Si un usuario pierde su clave para firmar documentos, basta con generar una nueva clave y un nuevo certificado.

Un sistema de archivado de claves se ocupa del almacenamiento seguro de las claves durante largos periodos de tiempo, mientras que un sistema de recuperación se ocupa de las operaciones necesarias para recuperar las claves. Los dos sistemas están fuertemente ligados. Habitualmente estos sistemas almacenan junto con la clave privada el certificado correspondiente.

Son varias las cuestiones que deben considerarse relativas al servicio de archivado y recuperación de claves: la ubicación de la base de datos de claves, el transporte de las claves, la protección de la base de datos de las claves, los procedimientos para recuperar claves, histórico de claves y custodia de claves.

El sistema de archivado y recuperación de claves puede ser un nuevo componente de la PKI o estar íntimamente ligado con algún componente ya existente. Además, dependiendo de las políticas y la implementación de la PKI el sistema de archivado deberá interactuar de diferentes modos con los restantes componentes de la PKI. Si se usa un sistema central de generación de claves, éste debe ser capaz de replicar las claves y transmitirlas de modo seguro al sistema de archivado o bien el sistema de archivado puede formar parte del propio sistema de generación de claves. Si el par de claves se genera en el sistema de la entidad final, la aplicación que genera las claves debe conectarse al sistema de archivado y transmitir la clave privada de forma segura. Si el par de claves se genera dentro de una smart card no puede transmitirse a un sistema de archivado. Para transmitir la clave privada de forma segura se usan distintos protocolos (PKCS #12, CMP, SSL, TLS...). En algunos sistemas la misma CA que genera el certificado se ocupa de almacenar la clave privada en una base de datos propia.

Una vez las clave privada y el certificado han alcanzado el sistema de archivado deben protegerse adecuadamente durante su almacenamiento. Para ello pueden usarse dispositivos hardware resistentes a manipulación, aunque es más habitual usar bases de datos cifradas. Una técnica habitual consiste en generar una clave simétrica aleatoria, cifrar la clave privada con dicha clave simétrica y cifrar la clave simétrica a su vez con la clave pública de uno o varios agentes de recuperación. Finalmente, se salva en la base de datos la clave privada cifrada y la clave simétrica cifrada con una o más claves públicas de los agentes de recuperación. Las claves privadas de los agentes de recuperación se deben proteger adecuadamente en dispositivos hardware o en software.

La recuperación de una clave privada debe ser tan transparente y fácil como sea posible, ya que de lo contrario la carga de trabajo del helpdesk o los administradores sería muy elevada. Si un usuario olvida o pierde su smart card con la clave privada en uso, no hay más remedio que recuperarla del sistema de archivado. Por otro lado, la PKI va generando periódicamente nuevos pares de claves para cada usuario, esto hace que documentos o e-mails recientes se vuelvan inaccesibles, ya que fueron cifrados con el par de claves anterior. Para evitar tener que recurrir con demasiada frecuencia al sistema de recuperación de claves cada usuario suele guardar un *histórico de claves*. La recuperación de claves privadas más antiguas no será tan frecuente y por tanto la carga de trabajo asociada con su recuperación del sistema de archivado es menor.

El acceso al sistema de archivado y recuperación debe ser restringido ya que las claves privadas son muy valiosas. Una posible alternativa es que, tanto si las claves se almacenan en un dispositivo hardware resistente a manipulación como si se usa un sistema simplemente de software, se requiera que varios administradores estén presentes y cada uno de ellos proporcione una clave para acceder al sistema. El problema de esta solución es que es muy costosa ya que requiere varios administradores y la recuperación de la clave privada puede diferirse excesivamente en el tiempo. Otra posible alternativa es que la recuperación de la clave privada sea realizada por el propio usuario. Esto plantea cuestiones acerca de la seguridad y privacidad de las claves, pero en muchas implementaciones el uso de un esquema adecuado de autenticación de los usuarios lo hace posible. Independientemente del método de acceso implementado, el sistema de archivado y recuperación de claves debe registrar cuidadosamente sus accesos.

La noción de archivado y recuperación de claves no debe confundirse con la *custodia de claves*. La custodia de claves está relacionada con el almacenamiento de las claves por motivos de seguridad nacional y cumplimiento de la ley. La custodia de claves se ha encontrado con una fuerte resistencia por parte de corporaciones, individuos y grupos privados que han obligado al gobierno a modificar su política sobre el tema. La lucha que mantienen las organizaciones gubernamentales contra los cárteles de la droga o la pornografía infantil hace que periódicamente se reavive la polémica sobre la custodia de claves.

4.2.10. Actualización de claves de entidad final

El término *actualización de claves* se refiere al proceso de generar nuevos pares de claves para usuarios existentes. Conviene diferenciar el proceso de actualización de claves del proceso de *renovación de certificados*. La renovación de certificados consiste en extraer la clave pública de un certificado e introducirla en un nuevo certificado con un periodo de validez diferente. La renovación de un certificado es un proceso simple que no involucra un nuevo par de claves. No hay una justificación criptográfica para la renovación de certificados, al contrario la renovación de certificados complica la PKI ya que obliga a redistribuir los certificados más a menudo. La renovación de certificados se utiliza mayoritarimente en servidores Web y tiene por objeto tarificar.

Volviendo a la actualización de claves, hay algunas razones que obligan a

limitar el periodo de vida de un par de claves:

- Las posibilidades de comprometer una clave aumentan con su uso.
- Si se compromete una clave, las pérdidas potenciales aumentan con el periodo de uso de la clave.
- Los avances tecnológicos pueden hacer más vulnerable una clave.

Políticas de seguridad sólidas restringen la cantidad de datos expuestos si se compromete una clave privada y obligan a actualizar las claves. El proceso de actualización de las claves puede aprovecharse para cambiar el algoritmo de firma de los certificados, longitudes de las claves, información de políticas en los certificados,... Conviene que la actualización de claves sea un proceso automático y transparente para el usuario final, por ello los pares de claves se actualizan antes de que expiren y así evitar que un usuario experimente *denegación de servicio*.

La actualización de pares de claves destinados a cifrado está gobernada por la expiración de la clave pública. Cuando ha transcurrido el 70 % u 80 % del periodo de vida de la clave pública, se produce el primer intento de actualizar las claves. Si el usuario está off-line, el sistema continúa funcionando normalmente y reintentará más tarde la actualización. Las nuevas claves se usarán para todas las operaciones subsiguientes de cifrado. Hay un periodo de transición entre el primer intento de actualizar las claves y la fecha oficial de expiración de la clave pública. Este periodo permite a los otros usuarios adquirir el nuevo certificado y evitar problemas de denegación del servicio. Cuando se actualiza el par de claves de cifrado, la clave privada pasa a ser gestionada por el histórico de claves.

A diferencia de la actualización de pares de claves de cifrado, la actualización de pares de claves destinados a firmas digitales está gobernada por la expiración de la clave privada. Una vez se ha generado el nuevo par de claves de firma se debe destruir de forma segura la clave privada de firma anterior. Esta destrucción garantiza que nadie pueda usar la clave privada de nuevo forzándose el no-repudio. El periodo de validez del certificado antiguo debe extenderse mucho más allá que el instante final de validez de la clave privada antigua para que los usuarios puedan seguir validando firmas en documentos antiguos con la clave pública de verificación.

La actualización de claves asume que la entidad final ya posee unas claves y un certificado. Al disponerse ya de unas claves y un certificado, pueden

aprovecharse en el proceso de actualización. Por ejemplo, cuando el usuario solicita una nueva clave, firmar la solicitud con la clave antigua puede ser suficiente para la autenticación.

4.2.11. Actualización de claves de CAs

La actualización del par de claves de una CA es conceptualmente similar a la actualización del par de claves de una entidad final. Sin embargo, dado que hay muchas entidades finales que confían en los certificados de la CA se deben adoptar ciertas medidas para hacer la transición más sencilla. El estándar RFC4210 especifica un procedimiento para CAs raíz que emiten certificados autofirmados.

El procedimiento se basa en que la CA firma el nuevo certificado con la clave previa y el certificado antiguo con la clave nueva. El resultado es que una CA bajo proceso de renovación tiene 4 certificados:

- *OldWithOld* Se trata del certificado autofirmado original en el que la clave privada original se usó para firmar la clave pública original.
- *OldWithNew* La clave pública original firmada en un certificado con la nueva clave privada.
- *NewWithOld* La clave pública nueva firmada en un certificado con la clave privada original.
- *NewWithNew* Se trata del certificado autofirmado final en el que la clave privada nueva se usa para firmar la clave pública nueva.

Todas las entidades finales confían en el certificado antiguo, OldWithOld, cuando comienza el proceso de actualización de claves. El certificado NewWithOld avala la nueva clave pública generada con la clave pública antigua en la que se confía. Una vez se confía en la nueva clave, los usuarios que capturan el nuevo certificado NewWithNew confiarán en él. A partir de ese momento los certificados OldWithOld y NewWithOld ya no son necesarios. El periodo de validez del certificado NewWithOld comienza cuando se genera el par de claves nuevo y finaliza cuando expira la clave pública antigua. Durante ese periodo de validez los usuarios afectados deben reconocer las nuevas claves.

El certificado OldWithNew permite una transición más suave al nuevo par de claves. La mayoría de implementaciones usan un periodo de solapamiento

durante el cual se usan el certificado antiguo y el nuevo. Durante este periodo es posible recibir un certificado de una entidad final en el que la cadena de certificados se termina con el certificado antiguo. El certificado OldWithNew permite a un usuario que ya haya migrado a las nuevas claves seguir confiando en el certificado antiguo. El periodo de validez para el certificado OldWithNew comienza con la fecha en la que se generó el par de claves antiguo y finaliza con la expiración de la clave pública antigua.

4.2.12. Recuperación de certificados y CRLs

La recuperación de certificados se ocupa de la manera en que se consiguen los certificados de un repositorio por un usuario PKI. La recuperación de certificados se produce cuando se cifran datos para uno o más receptores o bien cuando se construyen caminos de certificados. La aplicación más común es la gestión de claves simétricas entre emisor y receptor o receptores. Se cifran los datos con una clave simétrica que a su vez se cifra con la clave pública de cada receptor que se extrae del certificado correspondiente. En el caso de las firmas digitales, el certificado de verificación de origen suele enviarse junto con los datos firmados formando parte del protocolo de intercambio. Así se evita tener que buscar el certificado requerido en un repositorio remoto. Los certificados disponen de la extensiones AIA (Authority Information Access) y CDP (CRL Distribution Points) donde se especifican las ubicaciones desde las que se pueden recuperar automáticamente los certificados y CRLs. El estándar X.509 se deriva del estándar de directorio X.500. El directorio X.500 es el método habitual de distribución de certificados y CRLs. Sin embargo, el directorio X.500 nunca ha llegado a implementarse completamente por lo que hay que usar otras alternativas. El estándar RFC2585 propone usar los protocolos http y ftp para publicar y recuperar certificados y CRLs y el estándar RFC4523 adapta servidores LDAPv2 para ser usados como repositorios PKIX.

4.2.13. Construcción y validación de caminos de certificación

Un camino de certificación es una cadena de certificados en la que el emisor del primer certificado es un punto de confianza y el sujeto del último certificado es la entidad final que se intenta validar. Este último certificado

contiene la clave pública que se usará para validar una firma o establecer una clave simétrica. Una aplicación PKI debe construir y validar un camino de certificación antes de usar la clave pública para validar una firma o establecer una clave simétrica.

Construcción de caminos de certificación

Los métodos para construir caminos de certificación dependen de la arquitectura de la PKI (jerárquica, malla,...). El cliente PKI se inicializa para reconocer caminos que comienzan con una o varias CAs. Estas CAs se conocen como *puntos de confianza*.

En el caso de CA única o lista de confianza simple el camino de certificación consta de un único certificado, el de la entidad final. En este caso degenerado no hay construcción de camino.

En el caso de las PKI jerárquicas, la construcción de los caminos de certificación comienza por el certificado de la entidad final. Este certificado tiene un *Issuer Name* y una extensión *Authority Key Identifier*. Juntos, estos valores ayudan a localizar el certificado de CA correcto. El campo *Issuer Name* permite localizar en el repositorio los certificados de la CA, mientras que la extensión *Authority Key Identifier* coincide exactamente con *Subject Key Identifier* del certificado de CA que interesa. Este proceso se repite hasta encontrar un certificado emitido por la raíz, el punto de confianza de la jerarquía. En las jerarquías el camino de certificación es predecible y la entidad final lo puede proporcionar como parte del protocolo.

En el caso de las arquitecturas en malla los certificados de las entidades finales son emitidos directamente por su punto de confianza. Por ello, diferentes usuarios pueden tener diferentes puntos de confianza y construirán diferentes caminos de certificación. Además, una CA puede tener varios certificados, cada uno de ellos emitido por una CA diferente que lleva a una sección de la malla diferente. La construcción del camino comienza en el punto de confianza y se dirige hacia el certificado de la entidad final. El *Authority Key Identifier* coincidirá con el *Subject Key Identifier* de varios certificados de CA. Se debe ir probando con dichos certificados de CA; si uno de esos certificados no lleva a un camino de certificación completo, se debe probar con el siguiente. En cierto modo, establecer un camino de certificación en una malla es similar a enrutar un paquete a través de Internet. En una malla puede haber más de un camino entre un punto de confianza dado y una entidad final; generalmente se

usa el primer camino válido encontrado. La utilización de cache para guardar información de caminos puede mejorar significativamente el rendimiento.

En una arquitectura de lista de confianza extendida tanto puede haber PKIs jerárquicas como de malla. Un algoritmo ampliamente usado en este caso asume que el certificado de la entidad final pertenece a una jerarquía y utiliza el algoritmo jerárquico. Si el camino construido llega a uno de los puntos de confianza, ya se tiene un camino. Si no, se intenta construir un camino desde cada uno de los puntos de confianza hasta la raíz de la porción jerárquica previamente encontrada. De nuevo, el uso de caches en esta arquitectura puede comportar importantes mejoras de la eficiencia. Las implementaciones que soportan este tipo de arquitectura PKI son muy flexibles y funcionan en cualquier arquitectura de PKI.

Arquitecturas PKI que incluyen cross-certificados tienen muchas similitudes con mallas y listas de confianza extendida. Muchas implementaciones aplican el algoritmo jerárquico hasta que se encuentran múltiples emisores del certificado, entonces usan el algoritmo de malla comenzando en el punto de confianza. Puesto que hay un único punto de confianza, la construcción es más simple que en el caso de la arquitectura de lista de confianza extendida.

En la arquitectura de CA puente, hay un único cross-certificado que une la PKI nativa con todas las PKI restantes. Esta simplificación ayuda en la construcción de caminos. Cuando se puentean PKIs simples y jerárquicas la construcción del camino es ligeramente más complicada que en una PKI jerárquica. Se usa el algoritmo jerárquico hasta que falla, en ese momento sólo hay un cross-certificado disponible. Finalmente, localizar el cross-certificado emitido a la CA puente por el punto de confianza local es sencillo. Cuando se puentean PKIs en malla, la construcción de caminos en las mallas es compleja. Sin embargo, puesto que cada una de las PKIs remotas maneja un espacio de nombres diferente, las suposiciones acerca del cross-certificado apropiado son usualmente correctas.

El estándar RFC4158 del PKIX proporciona una guía útil para la construcción de caminos de certificados independiente de la arquitectura PKI. El documento sugiere usar un algoritmo para la construcción de caminos de certificados basado en el algoritmo de recorrido en profundidad de un árbol. Las CAs y los certificados que constituyen una PKI compleja se pueden representar como un grafo. A su vez, ese grafo puede convertirse en un árbol debido a que no está permitido repetir certificados en un camino. Los autores del estándar consideran que este algoritmo proporciona un equilibrio entre simplicidad

de diseño e independencia de la arquitectura de la PKI. El estándar anima a los desarrolladores de aplicaciones PKI a soportar arquitecturas PKI en puente ya que son las más generales y engloban a las restantes arquitecturas. Una aplicación que construye caminos de certificados válidos para arquitecturas PKI en puente también incorporará toda la lógica necesaria para soportar otras arquitecturas menos complicadas (jerárquica, malla, híbrida,...). En resumen, el documento estudia algoritmos específicos y mecanismos para asistir a los desarrolladores de software de construcción de caminos de certificación.

Validación de caminos de certificación

Una vez se ha construido el camino de certificados, se debe proceder a su validación. Son varios los estándares que se ocupan de la validación de caminos de certificación. El estándar X.509 proporciona un procedimiento general, mientras que el estándar RFC5280 del PKIX adapta dicho procedimiento al entorno de Internet. En esta sección se dará una visión general del procedimiento de validación del camino de certificados de acuerdo con el estándar RFC5280.

El procesamiento del camino de certificación verifica la asociación entre la identidad del sujeto del certificado y la clave pública del certificado. La asociación está limitada por restricciones impuestas por la parte que verifica el certificado y restricciones contenidas en los propios certificados del camino de certificación.

Un camino de certificados que va a ser validado es una secuencia de certificados que satisface las siguientes condiciones:

- Un punto de confianza emitió el primer certificado.
- El último certificado fue emitido a la entidad final y contiene la clave pública de interés.
- Los campos *issuer name* y *subject name* de los certificados forman una cadena. Para todos los certificados de la cadena, excepto el primero y el último, el *issuer name* coincide con el *subject name* del certificado anterior y el *subject name* coincide con el *issuer name* del siguiente certificado.
- Los certificados no han expirado.

Este conjunto de condiciones es necesario pero no suficiente para que la cadena sea válida. El diagrama de flujo del algoritmo de validación se muestra en la figura 4.1 y tiene los siguientes pasos:

- Inicialización
- Validación básica de un certificado
- Preparar el siguiente certificado
- Terminación

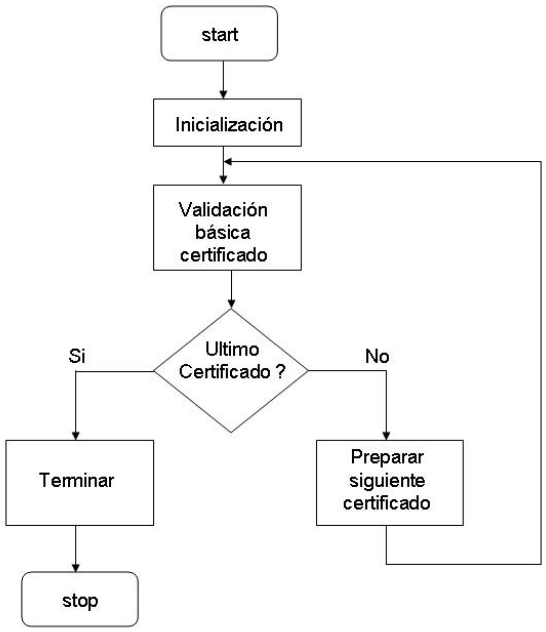

Figura 4.1: Validación camino de certificados

El algoritmo tiene unas entradas de información:

- Camino de certificados a evaluar
- Conjunto de identificadores de políticas válidos
- La información del punto de confianza, a menudo un certificado *self-signed*

- Un indicador de si se admite mapeo de políticas en el camino
- Un indicador de si se requieren identificadores de políticas explícitos en los certificados
- Un indicador de si se permite el valor especial *any-policy* en los certificados

La fase de inicialización establece unas variables de estado basadas en las entradas precedentes. Las variables de estado se usan para el seguimiento de varias restricciones a medida que la validación procede a través de la cadena de certificados. Hay cuatro grupos de variables de estado.

El primer grupo de variables de estado se ocupa de la información necesaria para verificar las firmas digitales. Estas variables incluyen la clave pública, parámetros asociados con la clave pública y el algoritmo de firma digital. Los algoritmos de firmas digitales requieren parámetros, algunos de ellos bastante extensos. Debido a eso no se repiten en los certificados subordinados si son los mismos que en el certificado anterior. El segundo grupo de variables de estado se encarga del encadenamiento de nombres y de las restricciones de longitud de la cadena de certificados. El tercer grupo de variables de estado se responsabiliza de las políticas de los certificados. Supervisa los identificadores de políticas, los resultados del mapeo de políticas y si está permitido, si se requiere un identificador de política y si el identificador *any-policy* está permitido. Finalmente, el cuarto grupo de variables de estado considera los nombres. Mantiene un registro de los subárboles de nombres permitidos y excluidos para cada forma de nombre. Los nombres válidos deben estar contenidos en los subárboles permitidos y no deben estar incluidos en ninguno de los subárboles excluidos.

La siguiente fase consiste en la validación básica de un certificado. En esta etapa se realizan varias verificaciones. En primer lugar se confirma si el tiempo actual está dentro del periodo de validez del certificado. Seguidamente, se comprueba que el certificado no ha sido revocado. Las CRLs son el mecanismo más habitual usado en esta verificación. Posteriormente, con las variables de estado de firma digital y la clave pública del *issuer* se valida la firma digital. Usando el segundo grupo de variables de estado se confirma que el *issuer name* coincide con el *subject name* del certificado precedente. A continuación, mediante el uso del tercer grupo de variables de estado, se verifican las políticas del certificado. Si la extensión de políticas está presente, verificar

que su valor está dentro de las políticas permitidas. Si la extensión contiene el identificador *any-policy*, verificar que está permitido. Si no hay extensión de políticas, comprobar si se requiere una política explícita. El último grupo de variables de estado se usa para verificar las restricciones de los nombres. Se debe comprobar que el *subject name* está en un subárbol permitido de DN (Distinguished Name) X.500 y no está dentro de los subárboles excluidos. Esta última validación debe extenderse también a las distintas formas de nombres alternativos del sujeto del certificado.

Si cualquiera de las comprobaciones básicas de un certificado falla, el camino de certificación es inválido. Si el certificado supera todas las verificaciones, el siguiente paso del algoritmo se determina por la posición del certificado que acaba de verificarse en la cadena de certificados. Si el certificado es el último de la secuencia se efectúa la terminación; si no, se realiza la preparación para el siguiente certificado de la secuencia.

La preparación para el siguiente certificado realiza diversas funciones. En primer lugar verifica que se trata de un certificado de CA, que contiene una clave pública de firma y que la extensión *key usage* permite firmar certificados. Luego se comprueba que no se ha superado la máxima longitud del camino de certificados. A continuación actualiza las variables de estado de firma digital. Establece los valores de la clave pública, parámetros asociados con la clave y el algoritmo de firma digital de acuerdo con el contenido del campo *subject public key*. Los parámetros del algoritmo de firma también pueden heredarse de un certificado anterior si no están explícitamente especificados en el certificado actual. El paso siguiente consiste en actualizar las variables de estado que se ocupan del encadenamiento de nombres. Establecer el *issuer name* esperado como el *subject name*. Si el *issuer name* y el *subject name* coinciden se trata de un certificado *self-issued*. Si el certificado no es *self-issued* incrementar el contador de certificados. No contabilizar los certificados *self-issued* permite a una CA actualizar sus claves sin invalidar el camino de certificados como resultado de un incremento en la longitud del camino de certificación. A continuación se realiza el mapeo de políticas y se actualizan las variables de estado de políticas. Para ello se convierten las políticas del emisor del certificado a las del sujeto del certificado. Basándose en las extensiones de políticas se establecen las variables de estado para indicar si los certificados subsiguientes deben contener identificadores de políticas explícitos, pueden contener el identificador *any-policy* o pueden contener mapeo de políticas. La siguiente función consiste en actualizar las variables de

estado de restricciones de los nombres. Si hay subárboles permitidos en el certificado, se establece la variable de estado a la intersección de los subárboles permitidos previos y los subárboles permitidos indicados en la extensión. Si hay subárboles excluidos en el certificado, se establece la variable de estado a la unión de los subárboles excluidos previos y los subárboles excluidos indicados en la extensión. La última función de esta fase consiste en reconocer y procesar cualquier extensión crítica presente en el certificado. Si cualquier extensión crítica no es reconocida el camino de certificados es inválido.

Si cualquiera de las comprobaciones de la fase de preparación del siguiente certificado falla, el camino de certificados es inválido. Si el certificado supera todas las comprobaciones se avanza al siguiente certificado en la secuencia y se pasa de nuevo a la etapa de validación básica.

La etapa de terminación completa el procesamiento del último certificado. En primer lugar actualiza las variables de estado de firma digital de acuerdo con las del último certificado. A continuación determina las políticas que se satisfacen calculando la intersección del conjunto de entrada de identificadores de políticas aceptables y las variables de estado de políticas. Si la intersección es vacía, la validación falla. Si la intersección no es vacía, establece la variable de estado de políticas a la intersección. Finalmente procesa las extensiones críticas adicionales. Si alguna extensión crítica no es reconocida la validación del camino de certificados falla.

Las salidas de un camino de certificación válido se derivan de las variables de estado e incluyen las políticas de certificados para el camino de certificación, la clave pública de la entidad final, parámetros asociados con la clave pública y el algoritmo de la clave pública.

Validación de CRLs

En el proceso de validación descrito en el apartado anterior es necesario determinar si cada certificado de la cadena ha sido revocado. El estándar RFC5280 describe los pasos necesarios para determinar si un certificado ha sido revocado temporalmente o permanentemente cuando el mecanismo utilizado son las CRLs. El algoritmo define un conjunto de entradas, un conjunto de variables y unos pasos que se deben llevar a cabo.

El algoritmo requiere dos entradas:

- El certificado. La combinación de los campos *serial number* e *issuer name* se usan para determinar si el certificado está en una CRL concreta.

La extensión *basic constraints* se usa para diferenciar un certificado de CA de uno de entidad final. Si están presentes en el certificado, el algoritmo usa las extensiones *CRL distribution point* y *Freshest CRL.*

- Un indicador que determina si se deben usar delta CRLs.

Basándose en las entradas, se establecen unas variables de estado. La variable de estado máscara de causas de revocación contiene el conjunto de causas de revocación soportados por las CRLs procesadas hasta el momento. Esta variable se incializa como conjunto vacío. La variable de estado *status del certificado* indica si el certificado ha sido revocado junto con la causa, si el certificado no ha sido revocado o bien un valor indeterminado. Esta variable se inicializa al valor de no revocado. Por último, la variable de estado máscara de causas provisional contiene el conjunto de causas de revocación soportadas por la CRL o Delta CRL que se está procesando.

El algoritmo revisa una o más CRLs hasta que se determina que el certificado ha sido revocado o bien hasta que se han cubierto todas las causas de revocación. El primer paso del algoritmo consiste en actualizar la cache local obteniendo una CRL completa, una CRL Delta o ambas. El segundo paso del algoritmo verifica el *issuer* de la CRL y el ámbito de la CRL (si es una CRL de certificados de CA o de certificados de EE). A continuación, se calcula la máscara de causas provisional y se comprueba que la máscara de razones provisionales incluye una o más razones que no están incluidas en la máscara de causas. También se obtiene y valida el camino de certificados para el *issuer* de la CRL completa. Se valida la firma de la CRL completa o CRL Delta. Si se encuentra en la lista de certificados de la CRL completa o CRL Delta una entrada que coincide con el *serial number* e *issuer* del certificado se actualiza la variable *status del certificado* con la causa de la revocación. Por último se actualiza el contenido de la máscara de causas con la unión de su valor previo y el valor de la máscara provisional. Si la máscara de razones ha cubierto todos los motivos o el *status del certificado* no es no revocado se devuelve el valor de la variable *status del certificado*. Si el *status del certificado* es no revocado se repite el proceso con las CRLs emitidas por el *issuer* del certificado.

Otros mecanismos

Construir y validar un camino de certificados para un certificado es un procedimiento complejo. Descargar toda o parte de esta carga de trabajo de

un cliente tiene implicaciones en la facilidad de despliegue y aceptación de una PKI. Es por ello que el grupo PKIX de IETF ha llevado a cabo esfuerzos de estandarización en esta dirección. Concretamente el estándar RFC2560 desarrolla el protocolo Online Certificate Status Protocol (OCSP) útil para determinar el estado de un certificado sin necesidad de CRLs. Por otro lado, el estándar RFC5055 describe el protocolo Server-Based Certificate Validation Protocol (SCVP) que permite a un cliente delegar en un servidor la contrucción y validación de caminos de certificación.

4.2.14. Revocación de certificados

La revocación de un certificado se ocupa de la cancelación de un certificado antes de que expire. La necesidad de revocar un certificado puede derivarse de diferentes causas: compromiso de la clave privada, cambio en la jerarquía de la empresa, finalización del empleo,.... El propio usuario puede iniciar la solicitud de revocación de su certificado. La solicitud puede hacerse on-line con una CA o RA, o bien off-line (llamada telefónica, presencialmente,...). En este último caso la RA realiza la solicitud en lugar del usuario. Los administradores autorizados también pueden iniciar el proceso de revocación cuando las circunstancias lo requieran.

4.2.15. Expiración de certificados

Los certificados tienen un periodo de validez que se determina en el instante de su emisión. Cuando un certificado expira pueden darse los 3 casos siguientes:

- Si la entidad final deja de formar parte de la PKI no se lleva a cabo ninguna acción.
- La renovación del certificado ocurre cuando la misma clave pública se usa en un nuevo certificado con un nuevo periodo de validez.
- La actualización del certificado ocurre cuando se genera un nuevo par de claves y se emite un nuevo certificado.

La renovación de un certificado se lleva a cabo si las circunstancias asociadas con la emisión del certificado original no han cambiado y si se considera que el par de claves es todavía criptográficamente sólido. Es deseable que las

firmas digitales producidas con un certificado que goza de unos determinados atributos puedan diferenciarse de las firmas digitales producidas con la misma clave pero diferentes atributos del certificado. A tal efecto se sugiere que se acoplen de forma segura la firma digital y el correspondiente certificado de verificación.

4.3. Protocolos de gestión

Para soportar todas las funciones del ciclo de vida se han definido protocolos estándar. En general, el propósito de los estándares es proporcionar una especificación común que pueda ser usada como un fundamento para la implementación. Sin embargo, la especificación puede ser incompleta y esto conlleva malas interpretaciones o interpretaciones erróneas que, a su vez, provocan dificultades de interoperabilidad. Los protocolos de gestión son necesarios para soportar las interacciones on-line entre los componentes de una PKI.

En esta sección se describirán los principales estándares de protocolos de gestión de una PKI. El grupo PKIX ha definido dos protocolos de gestión. En primer lugar el protocolo CMP (Certificate Management Protocol) definido en el documento RFC4210 que usa, a su vez, el estándar CRMF (Certificate Request Message Format) definido en RFC4211. En segundo lugar, el protocolo CMC (Certificate Management over CMS) definido en el documento RFC5272. Además de estos protocolos, Cisco Systems desarrolló el estándar SCEP (Simple Certificate Enrollment Protocol) para soportar la emisión de certificados a dispositivos de red de una manera escalable y usando tecnología existente.

A continuación se definen unos criterios que permiten comparar la bondad de los protocolos de gestión:

- *Completitud*. La completitud de un protocolo es función de los formatos de sus mensajes y de las secuencias de transacciones. Los protocolos deben definir formatos de mensajes de petición, respuesta, error, confirmación,... para cada una de las transacciones del ciclo de vida. También deben definir secuencias de mensajes para cada transacción, de lo contrario diferentes implementaciones no podrían interoperar.

- *Modelos de transacciones*. Un protocolo de gestión debe ser flexible

y soportar modelos de transacciones en los que intervengan dos (EE y CA) y tres (EE, RA y CA) partes.

- *Independencia del algoritmo*. El protocolo no debe estar ligado a ningún algoritmo criptográfico en particular. En general, una organización necesitará diferentes algoritmos para diferentes sistemas que además pueden ir cambiando con el tiempo. Este criterio es especialmente crítico cuando se habla de POP. Por ejemplo, el protocolo PKCS #10 sólo proporciona POP de la clave privada si se trata de una clave privada de firma, pero no si se trata de una clave privada dedicada a gestión de claves.

- *Complejidad de las transacciones y eficiencia*. Simplicidad y eficiencia son características deseables en un protocolo. Los protocolos deben alcanzar sus objetivos con el mínimo número de mensajes. Si una transacción requiere más mensajes que petición - respuesta, hay que mantener el estado de las diferentes transacciones lo que aumenta la complejidad de la implementación.

- *Extensibilidad*. Es imposible diseñar protocolos generales que se adapten a todas las circunstancias posibles. Los protocolos incorporan mecanismos que proporcionan flexibilidad y permiten ajustar el protocolo a entornos muy diversos. Por contra, estos mecanismos de flexibilidad originan problemas de interoperabilidad.

- *Archivado de mensajes*. Los protocolos de gestión deben permitir que RAs, CAs y EEs archiven pruebas de una transacción concreta. Por ejemplo, una EE puede necesitar demostrar que realizó una petición de revocación de su certificado. En algunos protocolos, los mensajes transmitidos permiten demostrar a los participantes que se comportaron correctamente. En otros protocolos, la información no está disponible en el formato adecuado.

- *Aprovechar código existente*. Este criterio es de índole meramente práctica. Si se aprovecha código existente se reduce el tiempo y el coste del desarrollo del software.

4.3.1. PKCS #10

PKCS #10, Certification Request Syntax Standard, no es propiamente un protocolo de gestión y por ello suele usarse conjuntamente con otros estándares (SSL, CMC,...). PKCS #10 describe la sintaxis de un mensaje de solicitud de certificado. En primer lugar se construye una estructura de datos consistente en el nombre distinguido (DN) del sujeto del certificado, una clave pública y un conjunto opcional de atributos. Seguidamente se firma la estructura de datos con la clave privada del sujeto. Por último, se forma la estructura de datos completa formada por la estructura de datos anterior, un OID que identifica el algoritmo de firma y la propia firma digital como se muestra en la figura 4.2.

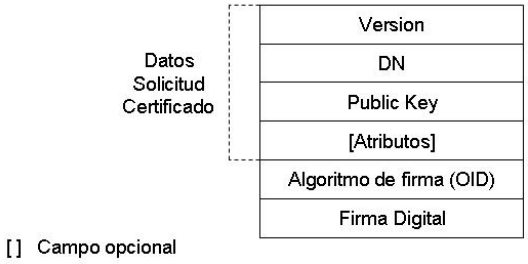

Figura 4.2: Formato mensaje PKCS #10

PKCS #10 tiene sus raíces en los estándares PEM (Privacy Enhanced Mail) del IETF. La firma prueba que el usuario que genera la petición tiene la clave privada correspondiente, pero no proporciona autenticación del origen ni integridad. El estándar no especifica el formato de la respuesta a este mensaje aunque menciona que una posibilidad es PKCS #7. Tampoco es-

pecifica cómo se debe llevar a cabo una revocación. El mensaje PKCS #10 no es independiente del algoritmo ya que la clave debe permitir firmas digitales. A pesar de todas estas limitaciones PKCS #10 es ampliamente usado por su simplicidad con otros protocolos como por ejemplo SSL y PKCS #7.

4.3.2. PKCS #7

PKCS #7 tampoco es un protocolo, sino una estructura de datos denominada *contentInfo* que protege mensajes criptográficamente. La estrucura de datos contentInfo tiene 2 campos: *contentType* y *content*. El contentType se especifica como un OID y hay 6 posibles: Data, SignedData, EnvelopedData, SignedAndEnvelopedData, DigestedData y EncryptedData. Según el OID del campo contentType será la estructura de datos content, es por ello que se dice que PKCS #7 es una estructura de datos parametrizada. La flexibilidad y complejidad de esta estructura de datos deriva de su naturaleza recursiva; el campo content puede contener a su vez otras estructuras de datos contentInfo. La figura 4.3 muestra la estructura de datos.

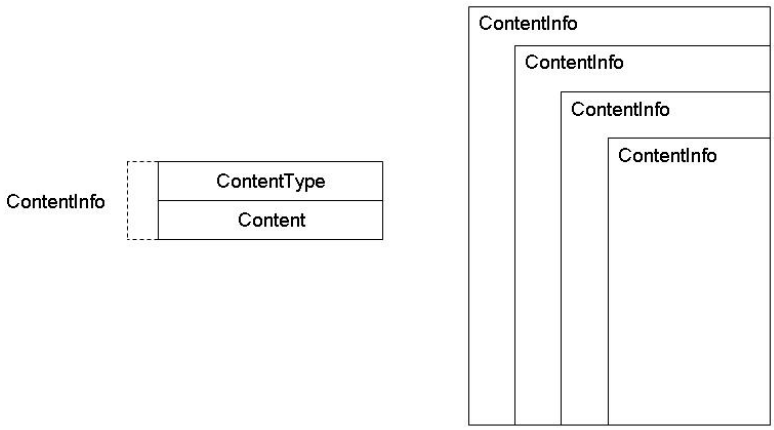

Figura 4.3: Estructura de datos PKCS #7

La estructura de datos más utilizada en PKIs es la SignedData. En ese caso el campo content incluye a su vez 6 campos: la versión, los algoritmos usados para generar las firmas, el contenido (contentInfo), certificados, CRLs e información de los firmantes. Los campos certificados y CRLs son opcionales. El campo de información de los firmantes consiste a su vez de 6 campos: la versión, el emisor y número de serie del certificado que contiene la clave pública para verificar la firma digital, el algoritmo de firma digital, la propia firma digital y atributos autenticados y no autenticados. Los atributos autenticados y no autenticados son opcionales. La firma digital se genera sobre el campo contenido y los atributos autenticados si los hay. De esta manera todas las firmas incluyen el campo contenido y cada firmante lo puede suplementar con los atributos autenticados. La figura 4.4 muestra una estructura SignedData encapsulando una solicitud de certificado PKCS #10.

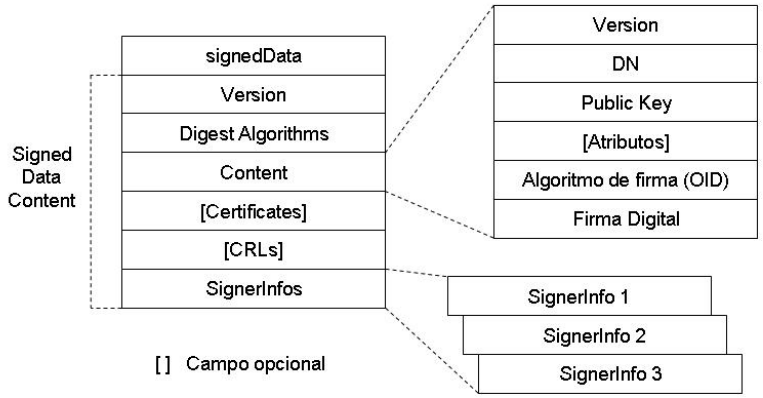

Figura 4.4: Estructura de datos PKCS #7 encapsulando solicitud de certificado PKCS #10

PKCS #7 y PKCS #10 pueden combinarse para soportar la emisión de certificados con o sin RA. Por ejemplo, Alice puede generar su propio par de claves pública y privada, generar una solicitud de certificado PKCS #10 y

entregarla en persona a la RA. La RA verifica los datos y la firma y genera un mensaje PKCS #7 para la CA. La CA autentica la RA a través de la firma PKCS #7, emite el certificado y lo devuelve a Alice o a la RA en forma de mensaje PKCS #7 firmado por la propia CA. Si Alice ya dispone de una clave privada para firmar no es necesaria la intermediación de la RA. En este caso, una vez Alice ha generado su propio par de claves, usa su clave privada de firma para generar la solicitud PKCS #7. La CA autentica Alice a través de la firma PKCS #7, emite el certificado con la nueva clave pública y se lo devuelve a Alice en forma de mensaje firmado PKCS #7.

PKCS #10 conjuntamente con PKCS #7 pueden usarse para solicitar certificados de un modo sencillo y extensible, pero no permiten revocar certificados ni proporcionan mensajes de error o confirmación. La POP sólo puede soportarse si se trata de una clave de firma. En definitiva se trata de un protocolo simple pero incompleto.

RSA cedió el control del protocolo PKCS #7 a la IETF pasando a denominarse *Crytographic Message Syntax*. La versión más actual se halla en el documento RFC5652.

4.3.3. CMP

Los primeros estándares que se usaron en el desarrollo de protocolos de gestión de certificados fueron PKCS #10 y PKCS #7. Estos protocolos presentaban varias dificultades. Eran incompletos, dependientes del algoritmo, inadecuados para archivar mensajes y además el propietario de los estándares era RSA. Todos estos inconvenientes ocasionaron que el grupo PKIX del IETF se decantara por el desarrollo de un estándar completamente independiente de PKCS #10 y PKCS #7 y que resolviera sus inconvenientes. Nacieron así los estándares RFC2510 (Certificate Management Protocol, CMP) y RFC2511 (Certificate Request Message Format, CRMF). En el año 2005 ambos estándares fueron revisados y sustituidos por los documentos RFC4210 y RFC4211 respectivamente. La parte referente a los protocolos de transporte del estándar RFC2510 se suprimió del documento RFC4210 y, en la actualidad, el grupo PKIX está elaborando un estándar titulado *Transport Protocols for CMP*. Este último documento describe dos posibles protocolos de transporte: TCP y HTTP; pero se declara firme partidario de HTTP.

El protocolo CMP se diseñó con unos objetivos concretos:

- Debe permitir la actualización regular de pares de claves sin afectar

otros pares de claves.

- El uso de la confidencialidad debe mantenerse en un mínimo para permitir su uso en entornos donde una fuerte confidencialidad podría causar problemas.
- El protocolo debe permitir la generación de pares de claves en cualquier componente de la PKI (RA, CA, EE).
- El protocolo debe ser independiente del algoritmo y permitir el uso de diferentes algoritmos criptográficos (RSA, DSA, SHA-1,...).
- El protocolo debe soportar la publicación de los certificados por cualquier componente de la PKI (RA, CA, EE).
- El protocolo debe soportar la producción de CRLs y la revocación de certificados.
- El protocolo debe soportar una variedad de mecanismos de transporte (TCP, HTTP,...).
- La responsabilidad última sobre la creación de certificados radica en la CA. Las RAs y EEs no pueden asumir que el contenido del certificado es el solicitado; una CA puede alterar campos y variar extensiones de acuerdo con su política operativa.
- El protocolo debe soportar una actualización de claves de CA grácil.
- Cuando una EE solicita un certificado conteniendo el valor de una clave pública, la EE debe ser capaz de demostrar la posesión de la clave privada correspondiente.

El formato del mensaje CMP se muestra en la figura 4.5. El mensaje tiene 4 componentes: *header*, *body*, *protection* y *extra certificates*. Los componentes *protection* y *extra certificates* son opcionales. El componente *protection* se usa para proteger la integridad del *header* y *body*. Transporta una firma digital o un código MAC. El campo *protection* se omite si *header* y *body* ya están protegidos de alguna otra manera. El campo *extra certificates* se usa para transportar certificados necesarios.

El *header* contiene el nombre del emisor, nombre del receptor, fecha y hora del mensaje y el algoritmo criptográfico para proteger el mensaje. El

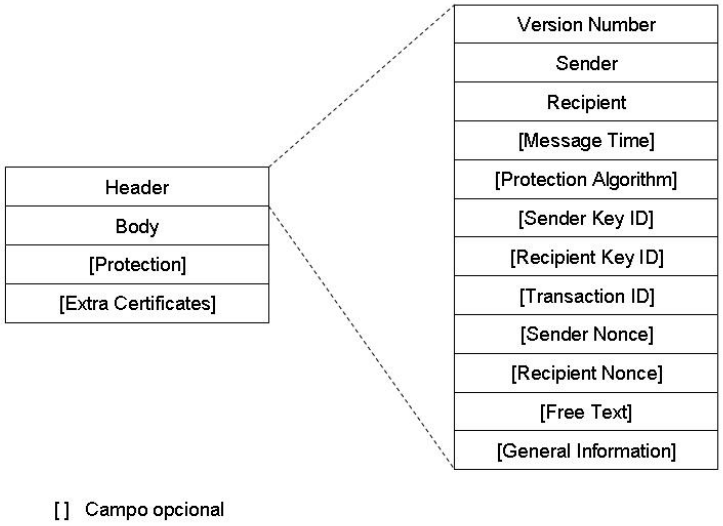

Figura 4.5: Estructura del mensaje CMP

header también puede contener otros campos opcionales como identificadores de claves, nonces, identificador de transacción,...

El campo *body* puede contener 27 tipos de mensajes diferentes contenidos en 20 estructuras de datos ASN.1 diferentes. Algunas estructuras de datos, particularmente las que hacen referencia a la solicitud de certificados, se especifican en el documento RFC4211 (CRMF). Las restantes estructuras se detallan en el documento RFC4210 (CMP). Los mensajes incluyen:

- *Mensajes de solicitud de certificado y respueta*. El protocolo soporta distintos tipos de solicitudes de certificados (inicial, cross-certificado, renovación de certificado,...) y sus respuestas.

- *Mensajes de solicitud de revocación y respuesta*. Estos mensajes se usan para solicitar la revocación de certificados y confirmar o denegar la solicitud de revocación.

- *Mensajes de solicitud de recuperación de clave y respuesta*. Estos mensajes solicitan y devuelven la clave privada para gestión de claves si la

PKI realizó una copia de seguridad.

- *Mensajes POP*. Mensajes usados para probar la posesión de claves privadas.

- *Mensajes para la distribución de certificados y CRLs*. Mensajes usados para anunciar la emisión de un certificado o CRL.

- *Otros mensajes*. El protocolo también incluye mensajes de confirmación, error,...

La mayoría de mensajes CMP están diseñados para transportar múltiples peticiones en un único mensaje. Esta característica permite a un usuario con 2 pares de claves (una para firmar y la otra para gestión de claves) enviar una única petición. También permite el procesamiento por lotes por las RAs cuando se deben procesar muchas peticiones o cuando las RAs operan principalmente offline.

CMP califica algunas funciones como obligatorias en el sentido de que todas las implementaciones de entidades finales, RAs y CAs deben ser capaces de proporcionar la funcionalidad descrita. Además, describe dicha funciones con cierto detalle. Las funciones son las siguientes:

- Inicialización de CA raíz.

- Actualización de claves de CA raíz.

- Inicialización de CA subordinada.

- Producción de CRL.

- Solicitud de información PKI.

- Cross-certificación.

- Inicialización de entidad final.

- Solicitud de certificado.

- Actualización de claves de entidad final.

Debido a la complejidad de los mensajes CMP, diferentes implementaciones CMP pueden no ser compatibles. Sin embargo, CMP define algunas funciones con suficiente detalle como para alcanzar la interoperabilidad. Estas funciones describen el flujo de mensajes y los campos obligatorios y opcionales de dichos mensajes y son las siguientes:

- *Cross-certificación en un sentido*. El protocolo de cross-certificación se usa cuando las CAs desean realizar la cross-certificación on-line. Este protocolo sólo puede usarse si las CAs pueden autenticar el origen de los mensajes, posiblemente a través de un mecanismo fuera de banda. La transacción requiere como mínimo dos mensajes: la CA solicitante presenta un mensaje *cross-certification request* y la otra CA responde con un mensaje *cross-certification response*. Los mensajes de confirmación son opcionales.

- *Certificación y registro inicial*. El protocolo de registro inicial se usa cuando una entidad final solicita su certificado inicial a una CA. En este protocolo la entidad final solicita la certificación de una clave pública generada localmente mediante un mensaje *initialization request*. La entidad final debe soportar la prueba de posesión (POP) de la clave privada. La CA responde con un mensaje *initialization response* conteniendo un certificado. La entidad final responde con un mensaje *certificate confirm* y la CA acaba el diálogo con un mensaje de confirmación.

- *Solicitud de certificado*. El protocolo de solicitud de certificado se usa cuando una entidad final que ya posee un certificado de firma válido solicita un nuevo certificado a su CA. Este certificado puede ser un certificado de gestión de claves o un nuevo certificado de firma digital. Con este protocolo la entidad final solicita el certificado con el mensaje *certificate request*. La entidad final debe soportar la prueba de posesión de la clave privada. La CA responde con un mensaje *certificate response* conteniendo el certificado. La entidad final responde a su vez con un mensaje *certificate confirm* y la CA acaba el diálogo con un mensaje de confirmación.

- *Solicitud de actualización de claves*. El protocolo de solicitud de actualización de claves lo usa una entidad final cuando actualiza el par de claves o el certificado correspondiente que ya posee. La entidad final solicita la certificación de la clave pública mediante el mensaje *key*

update request message. La CA responde con el mensaje *key update response* conteniendo el certificado. Finalmente la entidad final envía un mensaje de confirmación y la CA envía a su vez otro mensaje de confirmación para cerrar la transacción.

- *Solicitud y respuesta de información PKI*. En este protocolo la entidad final envía un *general message* a la RA o CA solicitando datos necesarios para operaciones posteriores. La RA o CA responde con un mensaje *general response*. Este protocolo no requiere mensajes de confirmación.

CMP es un protocolo completo. Dispone de un amplio rango de mensajes que abarcan todas las transacciones del ciclo de vida y es suficientemente flexible como para soportar transacciones entre 2 ó 3 participantes (EE, RA y CA). Además, dispone de muchas herramientas que permiten realizar POP de diferentes tipos de claves. Es un protocolo extensible, ya que incluye el campo *general information* en el *header* pensado explícitamente para esa misión. Otra ventaja es que, en general, todos los mensajes están firmados y eso permite archivarlos. Sin embargo, CMP es un protocolo relativamente complejo. En ocasiones no está claro qué mensaje aplicar y muchas transacciones requieren dos o más ciclos mensaje-respuesta lo que obliga a mantener el estado de las transacciones en el servidor y complica su implementación. Además, al ser CMP un protocolo completamente nuevo la sintaxis y el formato de los mensajes también lo son, todo esto hace que el desarrollo de componentes PKI CMP requiera mucho código nuevo.

4.3.4. CMC

La dificultad que entrañaba la implementación de CMP provocó la aparición de disensiones dentro del grupo PKIX. De repente se volvió crucial aprovechar el código que se había desarrollado para los antiguos protocolos existentes (PKCS #10 y PKCS #7) y simplificar al máximo el protocolo. Fruto de todo esto el grupo PKIX se escindió en dos bloques. Por un lado estaban los fabricantes partidarios de CMP que ya habían invertido en su desarrollo y, por otro, se hallaban los fabricantes que tenían mucho código desarrollado basado en PKCS #10 y PKCS #7. Cuando RSA decidió ceder el control de algunos estándares PKCS, entre ellos PKCS #10 y #7, al IETF se le allanó el camino para encontrar una solución. Se permitió la aparición de un segundo estándar

que compartiera con el primero (CMP) el formato de los mensajes de solicitud de certificados, proporcionase independencia de algoritmo e incluyese soporte para RAs. Este segundo protocolo usaría la nueva especificación CMS para proteger criptográficamente los mensajes. Nació así RFC2797, *Certificate Management over CMS* (CMC). CMC usa PKCS #10 para una solicitud de formato básica y CRMF para una solicitud de certificado más versátil. Además CMC depende de RFC2630, *Cryptographic Message Syntax*(CMS), para cifrar y firmar mensajes. CMS es la evolución de PKCS #7. En junio de 2008 se publicó una segunda versión de CMC, RFC5272, y la versión más reciente de CMS es el documento RFC5652 de septiembre de 2009. El documento RFC5272 es una revisión importante del RFC2797 y algunas secciones del estándar RFC2797 han generado nuevas especificaciones. En concreto la información referente al transporte de mensajes se ha trasladado al documento RFC5273 (CMC: Transport Protocols) y la información referente a las partes que se deben implementar se ha trasladado al documento RFC5274 (CMC: Compliance Requirements).

Algunos requisitos que se han tenido en cuenta para el diseño del protocolo son los siguientes:

- Debe estar basado tanto como sea posible en los estándares CMS, PKCS #10 y CRMF.
- Debe soportar la práctica habitual de la industria de una solicitud de certificado PKCS #10 y una respuesta PKCS #7 *certs-only*.
- Debe soportar S/MIME.
- Debe minimizar el número de pares mensaje-respuesta.
- Debe soportar formas de POP.
- La generación de las claves debe producirse en el cliente.

Una transacción de solicitud de certificado PKI se compone generalmente de un par mensaje-respuesta. En el caso más simple una solicitud se envía del cliente al servidor y el servidor responde enviando una respuesta al cliente. En casos más complicados, como una emisión de certificado diferida, se requiere más de un par mensaje-respuesta.

Esta especificación define dos tipos de solicitudes de certificados y dos tipos de respuestas. Los mensajes de solicitud son:

- Solicitud simple: una estructura PKCS #10.
- Solicitud completa: una o varias estructuras PKCS #10, CRMF u otras estructuras de mensajes envueltas en una estructura CMS formando parte, a su vez, de una estructura PKIData.

Las dos respuestas son:

- Respuesta simple: una estructura CMS SignedData *certs-only*.
- Respuesta completa: una estructura PKIResponse envuelta en una estructura CMS SignedData.

El protocolo no proporciona servicios para la renovación de certificados o renovación de claves. En su lugar se usa la solicitud-respuesta de certificados, excepto que la prueba de identidad se suministra a través de los certificados existentes. La especificación permite participar a RAs en el protocolo envolviendo las solicitudes PKI en un segundo envoltorio con los requisitos adicionales de la RA y pasando la nueva solicitud expandida a la CA. Otros servicios disponibles en esta especificación son la gestión de transacciones, detección de repeticiones (a través de nonces), emisión diferida de certificados, revocación de certificados y recuperación de certificados y CRLs.

CMC permite el uso de un mensaje PKCS #10 desprotegido por motivos de compatibilidad y también permite un mensaje CMS SignedData con contenido vacío para transportar certificados. Este último mensaje se llama *certs-only*. El resto de mensajes CMC se protegen mediante CMS. La especificación CMC define dos nuevas estructuras de datos: PKIData y PKIResponse. PKIData es esencialmente una solicitud de certificado y PKIResponse es el mensaje de respuesta de la CA. La mayoría de los mensajes CMC no requieren confidencialidad y se construyen encapsulando las estructuras PKIData y PKIResponse en una estructura CMS SignedData o en una estructura CMS AuthenticatedData. Cuando se requiere confidencialidad las estructuras SignedData o AuthenticatedData se envuelven a su vez en una estructura CMS EnvelopedData. CMC recomienda que a su vez la estructura CMS EnvelopedData sea recubierta de nuevo por una estructura CMS SignedData.

CMC utiliza controles. Los controles forman parte de los mensajes PKIData y PKIResponse y sirven para diversas funciones. Por ejemplo, hay controles para mantener el flujo de control de las transacciones, hay controles que

implementan nonces para evitar ataques de repetición, otros controles implementan POP,.... Cada control se codifica con un único OID seguido de los datos del control. La versión actual de CMC define 30 controles.

CMC no especifica completamente algunas transacciones por lo que no es un protocolo completo, pese a ello soporta modelos de transacciones en los que participan 2 ó 3 entidades (EE, RA, CA). También soporta POP para todos los tipos de claves. CMC es más complejo que PKCS #7 y PKCS #10 aunque más sencillo que CMP. Si bien hay pocos mensajes y la mayoría de transacciones se implementan en un único par mensaje-respuesta, hay muchos controles y no siempre está claro qué controles usar. CMC es también un protocolo extensible. Por otro lado, puesto que la mayoría de los mensajes van firmados son fácilmente archivables. En cuanto al aprovechamiento de código existente cabe decir que CMC aprovecha código de CMS y PKCS #10 pero hay muchas estructuras de datos nuevas que no se basan en código existente.

4.3.5. SCEP

Simple Certificate Enrollment Protocol (SCEP) es un protocolo desarrollado inicialmente por Cisco y Verisign para soportar la emisión segura de certificados a dispositivos de red de forma escalable utilizando tecnología existente. El protocolo utiliza PKCS #7, PKCS #10, HTTP y LDAP. El protocolo es un *Internet-Draft* de IETF titulado *Cisco Systems' Simple Certificate Enrollment Protocol*. La actual versión del documento es *draft-nourse-scep-20*. Se trata de un protocolo ampliamente difundido pero que dista mucho de ser completo. El documento anima a los desarrolladores a usar SCEP como un complemento de CMP o CMC. De hecho SCEP sólo soporta las siguientes operaciones:

- Distribución de claves públicas de CAs y RAs.
- Solicitud de certificados.
- Consulta de certificados.
- Consulta de CRLs.

El protocolo también soporta revocaciones a través de una password establecida durante la solicitud del certificado.

El cliente usa una operación GET de HTTP para obtener los certificados de la RA y CA. Una vez los ha obtenido debe contactar con el operador de la CA a través de algún mecanismo fuera de banda y verificar el hash de los certificados para asegurar la integridad de la operación.

La solicitud de certificados comienza cuando el cliente genera el par de claves pública y privada. El cliente genera entonces un certificado auto-firmado. Este certificado auto-firmado es necesario porque la solicitud de certificados usa PKCS #7 que, a su vez, requiere la existencia de un certificado. El cliente genera una solicitud de certificado PKCS #10 envuelta con una estructura PKCS #7 firmada y cifrada. Hay dos procedimientos para autenticar la solicitud del certificado. En el procedimiento de autenticación manual, la CA no responde hasta que el operador de la CA verifica la identidad del solicitante. En el procedimiento de autenticación a través de un secreto compartido, el solicitante suministra una password previamente distribuida para autenticar la solicitud. Esta password sirve después para soportar una posterior revocación.

Para revocar un certificado el administrador del dispositivo de red contacta con el operador del servidor y le comunica la password del procedimiento de solicitud de certificados. Si la password es correcta la CA revoca el certificado.

Para la consulta de certificados el protocolo propone dos alternativas. Una de ellas es usar LDAP y la otra es usar unos mensajes específicos del protocolo. Para la consulta de CRLs, si la CA soporta CRL Distribution Points (CDP) entonces se usa el mecanismo especificado en los CDPs. Si la CA no soporta CDPs se usan mensajes específicos del protocolo.

Globalmente SCEP es limitado y no soporta todas las actividades del ciclo de vida. Su rango de mensajes no es completo y no soporta todos los modelos de transacciones posibles. Tampoco es un protocolo particularmente adecuado para el archivado de mensajes. Por contra SCEP es simple y aprovecha muchas otras especificaciones (LDAP, HTTP, PKCS #7 y PKCS #10) lo que hace que requiera poco código nuevo.

4.3.6. Selección del protocolo

CMP es claramente el protocolo más completo ya que especifica más mensajes y transacciones que el resto de protocolos pero por contra es más complejo. CMC no es tan completo como CMP pero especifica suficientes mensajes y transacciones. Tiene la ventaja de aprovechar otros protocolos amplia-

mente distribuidos como PKCS #10. SCEP tiene un campo de aplicación muy limitado, mientras que PKCS #7 y PKCS #10 son claramente incompletos.

La mejor solución es una CA que soporte varios de estos protocolos. Como mínimo una CA debe soportar PKCS #7, PKCS #10, SCEP y CMP o bien CMC. Los sistemas clientes basta con que soporten uno de ellos para ser compatible con el producto.

4.4. Consideraciones sobre el mantenimiento

En esta sección del capítulo se realizarán algunas consideraciones generales sobre algunos aspectos relevantes del mantenimiento como son el personal encargado del soporte, la recuperación de desastres y otras cuestiones operativas.

4.4.1. Personal

Se requiere personal cualificado para administrar, operar y llevar a cabo todas las operaciones que se han detallado en el ciclo de vida de una PKI. La organización debe determinar el número y el nivel de conocimientos del personal de soporte que dependerá, entre otros factores, del tamaño de la PKI y del grado de externalización.

En general se requieren diferentes perfiles para operar una PKI:

- *Oficiales de seguridad*, responsables de hacer cumplir la política de seguridad de la empresa.
- *Operadores*, responsables de instalar sistemas, soportar usuarios, controlar backups,...
- *Administradores*, responsables, por ejemplo, de registrar usuarios, revocar certificados,...
- *Help desk*, soporte directo a usuarios (local o remoto).

Además, en las etapas de evaluación y diseño, deben considerarse otros perfiles como el de consultores especializados que ayuden a redactar o analizar los documentos de políticas y a desarrollar y documentar la estrategia de despliegue de la propia PKI.

Un aspecto importante que debe considerarse cuando se dimensiona el equipo de soporte es el impacto sustancial que tiene en el coste del mantenimiento de la PKI. Otro aspecto a considerar es que, debido a que la tecnología de las PKIs es relativamente reciente, el número de personas cualificadas en este sector es limitado. Y, aunque el número de expertos va aumentando, este tipo de personal puede ser relativamente difícil de retener. Si se va a usar personal de la propia organización, éste debe ser adecuadamente formado en su área particular de responsabilidad.

4.4.2. Prevención y recuperación de desastres

Cuando se diseña una PKI se debe desarrollar un plan para recuperación de desastres que asegure que, en el caso de fallo de algún componente de la PKI, éste se puede recuperar rápidamente con un efecto pequeño sobre la organización. Las razones habituales que hacen el plan de recuperación de desastres necesario son:

- Fallos de software, los fallos de software (corrupción o pérdida de ficheros) pueden provocar que algún servicio de la PKI (por ejemplo CA) no pueda arrancar. La manera tradicional de afrontar este tipo de problemas es mediante backups.

- Fallos de hardware, los fallos de componentes hardware pueden impedir, por ejemplo, que el servidor de una CA arranque. Para combatir estos errores puede usarse hardware redundante.

- Fallos de red, un error en la infraestructura de red puede impedir el acceso a información actualizada de CRLs, por ejemplo, e impedir que se puedan validar los certificados. La redundancia de componentes es una estrategia comúnmente usada para solucionar este tipo de problemas.

Junto a este tipo de desastres que se pueden dar en cualquier tipo de entorno, hay uno específico de las PKIs que consiste en el compromiso de la clave privada de una CA. Una organización debe implantar las medidas de prevención apropiadas para minimizar este tipo de riesgo y desarrollar los procedimientos y herramientas que ayuden una rápida recuperación si se produjese tal circunstancia.

El backup de una CA o RA presenta peculiaridades. Cuando se realiza el backup de una CA o RA se debe copiar su base de datos, sus pares de claves y todos los parámetros de configuración.

La base de datos de una CA o RA incluye detalles sobre cada certificado emitido y revocado por la CA o RA. Puesto que estos datos son altamente sensibles, se necesitan procedimientos especiales de backup para asegurar que los datos no se alteran o corrompen. Por ejemplo, si la base de datos falla y se restaura de una copia alterada, certificados previamente revocados pueden volver a ser válidos. Por este motivo, los datos de backup deben estar cifrados, protegidos con passwords o incluso firmados. Esto impide que puedan usarse las copias de seguridad si son robadas. Los medios de soporte del backup deben almacenarse en lugares protegidos, a prueba de incendios y con acceso restringido.

El backup de los pares de claves de la CA permite que cualquier certificado en uso emitido por la CA siga siendo válido cuando se haya restaurado la CA. Si el par de claves de una CA está siendo actualizado con un nuevo par de claves, el backup debe incluir los dos los pares de claves. Si la CA o RA utiliza un HSM, el backup de los pares de claves puede requerir un software especial del propio HSM.

Mención aparte merece el compromiso, la destrucción o la imposibilidad de usar la clave privada de una CA. Este suceso puede considerarse desastroso por la confianza que un grupo, potencialmente elevado, de entidades PKI depositan en la CA. En general, es peor el compromiso de la clave de una CA que la simple imposibilidad de usarla. En ambos casos se requiere el restablecimiento de la confianza en una nueva clave, pero el compromiso acarrea además la pérdida de confianza en los certificados que ha emitido la CA. Si alguien accede a la clave privada de una CA puede emitir certificados falsos, también puede emitir cross-certificados que pueden extender la confianza a una CA que pertenezca a un competidor,... Las implicaciones de seguridad que tiene el compromiso de la clave privada de una CA son extremas y por ello se considera un suceso desastroso.

La primera acción que se debe llevar a cabo para recuperar la PKI del compromiso de la clave privada de una CA es notificar el problema a la comunidad de usuarios. Esto no siempre es posible. Por ejemplo, es imposible informar a los millones de usuarios de PCs que tienen la clave pública de una CA implantada en su navegador que la clave privada correspondiente ha sido comprometida. En otros entornos, la notificación puede realizarse inclu-

yendo el certificado correspondiente en una CARL (Certification Authority Revocation List), aunque esté firmada por la propia clave privada de la CA comprometida. El problema de las CARLs es que buena parte del software PKI existente no soporta el tratamiento de CARLs. En ocasiones, no queda más remedio que usar e-mail u otros mecanismos fuera de banda para comunicar el compromiso.

En la práctica se toman tantas medidas preventivas como sea posible para evitar el compromiso o pérdida de uso de una clave privada de una CA. Por ejemplo, se usan dispositivos criptográficos de alta calidad (FIPS 140 de nivel 3 ó 4) para proteger la clave privada de una CA. Aún así, si se produce el compromiso hay algunas medidas que alivian la situación. Conocer la comunidad de usuarios. Fomentar el uso de CARLs. Usar claves con periodos de validez razonables, ya que periodos de validez cortos minimizan el perjuicio producido. Implantar un mecanismo automático de actualización de claves porque facilita usar periodos de validez más breves. En cualquier caso, los administradores de una PKI deben dedicar el tiempo necesario para prevenir y preparar el compromiso de una clave privada de CA.

Una vez se ha producido el compromiso de la clave privada de la CA, no hay más remedio que reiniciar la PKI desde cero para la comunidad de usuarios directos de la CA. Se debe generar un nuevo par de claves de la CA, implantar la nueva clave pública de la CA en los usuarios directos a través de un mecanismo seguro y volver a generar los certificados necesarios. Para los usuarios directos de esa CA se debe reconstruir la PKI como si nunca hubiera existido.

4.4.3. Consideraciones operativas

La confianza que se puede otorgar a un certificado depende de la seguridad del servidor de certificados que lo emitió. Los documentos de políticas deben detallar cómo administrar el servidor de forma segura.

Entre las actividades que se deben llevar a cabo periódicamente está la actualización del servidor con los parches del sistema operativo y de seguridad más recientes.

Además, el acceso a la CA debe estar restringido al menor número de administradores posible y éstos deben autenticarse al servidor CA antes de efectuar cualquier operación privilegiada. Si los administradores son remotos, además de autenticarse se debe cifrar la sesión. Cualquier acción de cualquier

administrador debe quedar convenientemente registrada.

Algunas leyes obligan a almacenar ciertos documentos durante un determinado número de años. En estos casos puede ser necesario archivar los certificados de verificación correspondientes. Estos archivos deben ser seguros de modo que no se puedan reemplazar certificados y alterar documentos. Se debe determinar cuánto tiempo se deben almacenar los certificados, cómo se han de proteger los certificados y los procedimientos para acceder a ellos.

El acceso a las claves privadas de las CAs debe estar protegido. Además del acceso convencional a las claves, debe haber un acceso alternativo en el caso, por ejemplo, de que algún administrador abandone la organización inesperadamente.

A medida que aumentan los usuarios que usan la PKI hay que simplificar los procedimientos para hacerles llegar los certificados raíz en los que confiarán. También puede ser necesario ajustar los periodos de publicación de las CRLs o decidir sobre la conveniencia de usar un servidor OCSP. Una PKI está en continua evolución y continuamente deben implantarse mejoras.

Capítulo 5

DNI electrónico y desarrollos futuros

5.1. Introducción

Este capítulo tiene dos vertientes. Por un lado muestra un ejemplo de aplicación de las PKIs: el DNI electrónico (DNI-e) y, a continuación, examina las tendencias actuales en la evolución de las PKIs y realiza algunas reflexiones sobre su viabilidad futura.

En cuanto al DNI-e se comienza analizando el marco legal. Seguidamente se describen las características físicas del documento y electrónicas del chip. A continuación se detalla el contenido del chip y el perfil de los certificados. Posteriormente se describe, a grandes rasgos, algunas de las características significativas de la PKI que soporta el DNI-e según el documento de *Declaración de Prácticas y Políticas de Certificación* (DPC). Se concluye la sección dedicada al DNI-e con información relativa al uso del DNI-e.

5.2. DNI electrónico

El DNI-e acredita física y electrónicamente la identidad y permite la firma electrónica de documentos.

Las *cédulas personales*, expedidas por las Diputaciones Provinciales a partir de 1931, fueron el origen de los actuales documentos nacionales de identidad (DNI) en España. El DNI se implanta con caracter obligatorio para los mayores de 16 años en 1944. A partir de 1990 comenzó el proceso de digi-

talización de los archivos del DNI. Como consecuencia de la rápida implantación de las redes de comunicación en nuestra sociedad surge la necesidad de implantar el DNI-e con un formato mejorado. Pasa de ser una simple tarjeta de identificación a tratar de ser un medio que impulse la Administración Electrónica en España, gracias a los nuevos servicios que incorpora en su chip. El proyecto del DNI-e comenzó en el año 2002. El primer ejemplar válido de DNI-e se expidió en Abril del 2006 en la comisaría provincial de Burgos. En la actualidad se emiten 30.000 documentos diarios en las casi 300 comisarías de todo el territorio español. El proyecto está dirigido por la Dirección General de la Policía y la Guardia Civil y lo ha lleva a cabo una unión temporal de empresas formada por Telefónica, Indra y Software AG. En la actualidad (inicios del 2011) se llevan expedidos 20.000.000 de DNI electrónicos.

Buena parte de la información de las secciones subsiguientes relativa al DNI-e se ha extraído del documento de *Política de Certificación* de la web *http://www.dnielectronico.es*. Este documento recoge la *Declaración de Prácticas y Políticas de Certificación* (DPC) que rige el funcionamiento y operaciones de la Infraestructura de Clave Pública de los Certificados de Identidad Pública y Firma Electrónica del Documento Nacional de Identidad. El documento se ha estructurado conforme a lo dispuesto por el grupo de trabajo PKIX del IETF en su documento de referencia RFC 3647 (Noviembre de 2003). Para dotar de uniformidad y facilitar la lectura y el análisis del documento se incluyen todas las secciones establecidas en el documento RFC 3647. Se ha incluido un epígrafe adicional dedicado a la protección de datos de carácter personal para dar cumplimiento a la normativa española en la materia.

5.2.1. Legislación

En esta sección se describe la legislación europea y española relativa al DNI-e. Se tratan diferentes aspectos como las condiciones que deben cumplir los prestadores de servicios de certificación y los requisitos que deben darse para que la firma con el DNI-e sea equivalente a la firma manuscrita.

Directiva europea de firma electrónica (1999/93/CE)

Para que la comunicación y el comercio electrónico se desarrollen adecuadamente son necesarios servicios de autenticación de datos y firmas electrónicas. Las diferencias normativas en materia de reconocimiento de firmas

electrónicas y acreditación de los proveedores de servicios de certificación entre los estados miembros de la Comunidad Económica Europea puede dificultar gravemente tanto el comercio como la comunicación electrónicas. El 13 de diciembre de 1999 el Parlamento Europeo aprobó la Directiva Europea 1999/93/CE con la intención de crear un marco armonizado para la prestación del servicio de firma electrónica y otros servicios relacionados como los servicios de certificación.

La directiva desarrolla un marco para los servicios de firma electrónica y autenticación, pero no de confidencialidad de la información. La directiva también promueve la interoperabilidad de los productos de firma electrónica para garantizar la libre circulación de bienes y servicios en el mercado interior europeo. Finalmente, la directiva introduce algunos nuevos conceptos como la *firma electrónica avanzada*, los *certificados reconocidos*, los *dispositivos seguros de creación de firma* y la *firma cualificada*.

Se entiende por *firma electrónica avanzada* una firma electrónica que cumple los requisitos siguientes: a) estar vinculada al firmante de manera única; b) permitir la identificación del firmante; c) haber sido creada utilizando medios que el firmante puede mantener bajo su control exclusivo; d) estar vinculada a los datos a que se refiere de modo que cualquier cambio ulterior de los mismos sea detectable.

Se entiende por *certificado reconocido* un certificado que cumple los requisitos del anexo I y es suministrado por un proveedor de servicios de certificación que cumple los requisitos establecidos en el anexo II. El anexo I establece los distintos campos que forman un certificado reconocido. El anexo II detalla una serie de requisitos a los prestadores de servicios de certificación (disponer de un servicio rápido y seguro de revocación de certificados, comprobar la identidad y otros atributos específicos de la persona a la que se expide el certificado, emplear personal con la experiencia y cualificación necesarias, utilizar sistemas y productos fiables que estén protegidos contra toda alteración y que garanticen la seguridad técnica y criptográfica de los procedimientos,...).

Se entiende por *dispositivo seguro de creación de firma* un dispositivo de creación de firma que cumple los requisitos establecidos en el anexo III. El anexo III establece que los dispositivos seguros de creación de firma garantizan, por medios técnicos y de procedimiento adecuados, que: a) los datos utilizados para la generación de firma sólo pueden producirse una vez en la práctica y se garantiza razonablemente su secreto; b) existe la seguridad ra-

zonable de que los datos utilizados para la generación de firma no pueden ser hallados por deducción y la firma está protegida contra la falsificación mediante la tecnología existente en la actualidad y c) los datos utilizados para la generación de firma pueden ser protegidos de forma fiable por el firmante legítimo contra su utilización por otros. Además los dispositivos seguros de creación de firma no alterarán los datos que deben firmarse ni impedirán que dichos datos se muestren al firmante antes del proceso de firma.

Finalmente, se entiende por *firma cualificada* una firma electrónica avanzada basada en un certificado reconocido y creada por un dispositivo seguro de creación de firma. Será admisible como prueba en procedimientos judiciales.

El EESSI (European Electronic Signature Standarization Initiative) nace con el objetivo de analizar las necesidades futuras de estandarización para soportar la directiva. Esta organización recomienda seguir el estándar RFC 3647 en cuanto a las políticas y la declaración de prácticas de certificación. También recomienda seguir el estándar BS 7799 del BSI (British Standards Institution) en cuanto a la gestión segura de la información. Este estándar se ocupa de que la información sea manipulada en los procesos de una manera adecuada a la política de seguridad. Asegura que, en todo momento, la información tiene la suficiente confidencialidad e integridad. En cuanto a los dispositivos de creación y verificación de firma recomienda seguir los estándares FIPS 140 o bien ISO 15408. Actualmente la tecnología más comúnmente usada para implementar un dispositivo seguro de creación de firma es una tarjeta inteligente en combinación con un lector. La tarjeta inteligente contiene la clave privada protegida con un PIN aunque la identificación biométrica se considera más segura y fácil de usar. En un futuro próximo, la biometría y, más concretamente, la identificación por huella dactilar reemplazarán el uso del PIN. Otros dispositivos como PDAs, teléfonos móviles con tarjetas SIM y tarjetas PCMCIA pueden ofrecer niveles de seguridad similares.

La ley de firma electrónica (Ley 59/2003)

El Real Decreto Ley 14/1999, de 17 de Septiembre, sobre firma electrónica, fue aprobado para incorporar rápidamente las nuevas tecnologías de seguridad de las comunicaciones electrónicas a las empresas, ciudadanos y administraciones públicas españoles. Por un lado, fomenta el uso de las transacciones electrónicas sobre Internet y, por otro, incorpora al ordenamiento legal

español la directiva 1999/93/CE del parlamento europeo incluso antes de su publicación en el Diario Oficial de la Comunidad Europea. Finalmente, el 19 de Diciembre de 2003 se publica la ley española 59/2003 de firma electrónica.

La ley se estructura en 6 títulos. El primer título se ocupa del ámbito de aplicación de la ley, efectos y empleo de la firma electrónica ante las administraciones públicas y el acceso a la actividad de prestación de servicios de certificación. El título segundo regula los certificados electrónicos y el DNI-e. El tercer título regula la actividad de los prestadores de servicios de certificación y establece las obligaciones a las que están sujetos. El título cuarto se refiere a los dispositivos de creación y verificación de firma electrónica y el procedimiento que ha de seguirse para obtener sellos de calidad en la actividad de prestación de servicios de certificación. Los títulos quinto y sexto fijan los regímenes de supervisión y sanción de los prestadores de servicios de certificación.

La ley traslada los conceptos de la Directiva europea de certificados reconocidos, firma electrónica reconocida o cualificada y dispositivo seguro de creación de firma. La ley define una clase particular de certificados electrónicos denominados *certificados reconocidos* que son los certificados electrónicos que se han expedido cumpliendo unos requisitos de contenido, de procedimientos de comprobación de la identidad del firmante y de fiabilidad y garantía de la actividad de certificación electrónica. Los certificados reconocidos permiten la *firma electrónica reconocida* que se define siguiendo las pautas impuestas en la Directiva 1999/93/CE como la firma electrónica avanzada basada en un certificado reconocido y generada mediante un dispositivo seguro de creación de firma. A la firma electrónica reconocida la ley le otorga la equivalencia funcional con la firma manuscrita respecto de los datos consignados de forma electrónica.

Según la ley, se denomina *prestador de servicios de certificación* la persona física o jurídica que expide certificados electrónicos o presta otros servicios en relación con la firma electrónica. La acreditación de un prestador de servicios de certificación es el procedimiento voluntario por el que una entidad cualificada pública o privada emite una declaración a favor de un prestador de servicios de certificación que implica un reconocimiento del cumplimiento de requisitos específicos en la prestación de los servicios que se ofrecen al público. La acreditación de un prestador de servicios de certificación podrá ser solicitada por éste y podrá ser llevada a cabo, entre otras, por entidades de certificación reconocidas por una entidad de acreditación designada de acuer-

do con lo dispuesto en la Ley 21/1992. En los procedimientos de acreditación podrán usarse normas técnicas u otros criterios de certificación adecuados. En caso de utilizarse normas técnicas, se emplearán preferentemente aquellas que gocen de amplio reconocimiento aprobadas por organismos de normalización europeos y, en su defecto, otras normas internacionales o españolas. La acreditación de un prestador de servicios de certificación no será necesaria para reconocer eficacia jurídica a una firma electrónica.

El Ministerio de Ciencia y Tecnología controlará que los prestadores de servicios de certificación cumplen los requisitos exigidos por la ley. Para ello realizará las actuaciones inspectoras que sean necesarias y podrá recurrrir a entidades independientes que le asistan en dicha función. También supervisará el funcionamiento del sistema y los organismos de certificación de dispositivos seguros de creación de firma electrónica. Los prestadores de servicios de certificación, la entidad independiente de acreditación y los organismos de certificación tienen la obligación de facilitar al Ministerio de Ciencia y Tecnología toda la información y colaboración precisas para el ejercicio de sus funciones.

ASIMELEC se constituyó como entidad certificadora de prestadores de servicios de certificación en el año 2005. En la actualidad distingue los siguientes tipos de certificados para la firma electrónica: certificado normalizado y certificado reconocido. Un certificado normalizado es aquel emitido a un prestador que garantiza el cumplimiento de la norma *ETSI TS 102 042* en lo relativo a las prácticas de certificación; mientras que un certificado reconocido es aquel emitido a un prestador que satisface todos los requisitos necesarios para proveer certificados reconocidos, conforme a la ley de firma electrónica (L59/03).

Los servicios de certificación pueden ser prestados tanto por entidades públicas como privadas. Entre las entidades públicas destacan CERES-FNMT (toda España), CATCert (Cataluña) e Izenpe (País Vasco). Entre las entidades privadas cabe nombrar Camerfirma y Firmaprofesional.

La certificación de dispositivos seguros de creación de firma es el procedimiento por el que se comprueba que un dispositivo cumple los requisitos establecidos por la ley. La certificación podrá ser solicitada por los fabricantes o importadores de dispositivos de creación de firma y se llevará a cabo por las entidades de certificación reconocidas por una entidad de acreditación designada de acuerdo con lo dispuesto la ley 21/1992. En los procedimientos de certificación se usarán normas técnicas aprobados por la Unión Europea

o el Ministerio de Ciencia y Tecnología. Los certificados de conformidad de los dispositivos seguros de creación de firma serán modificados o revocados cuando se dejen de cumplir las condiciones establecidas para su obtención y los organismos de certificación asegurarán la difusión de las decisiones de revocación de certificados de dispositivos de creación de firma. Según decisión de la Comisión Europea, la *CWA 14169* es la norma que goza del reconocimiento general para productos de firma electrónica de los estados miembros.

En la actualidad se están desarrollando varios proyectos emblemáticos de aplicación de la firma electrónica entre los que cabe destacar: DNI-e, Pasaporte electrónico, Tarjeta Sanitaria electrónica, Contratación y Facturación electrónica, e-Administración y Tacógrafo digital.

Posteriormente se han emitido otras leyes relacionadas con la sociedad digital. La ley 56/2007, de 28 de diciembre, de Medidas de Impulso de la Sociedad de la Información, pretende estimular la sociedad digital con una serie de medidas de obligado cumplimiento en sectores de gran influencia en la actividad económica. Obliga a las grandes empresas a facilitar medios de comunicación telemáticos con sus clientes basados en certificados reconocidos y, más concretamente, en el DNI-e; prevé la facturación electrónica; facilita la actividad económica por Internet (regulación de subastas electrónicas entre empresas, flexibilización de las obligaciones relativas a las comunicaciones comerciales y a los requisitos para la contratación por vía electrónica,...); fomenta la seguridad en Internet; extensión de la conectividad de banda ancha;.... Finalmente, la ley 11/2007 de Acceso de los Ciudadanos a los Servicios Públicos pretende una administración pública más efectiva y eficiente, además de reconocer a los ciudadanos nuevos derechos en su relación con las administraciones públicas. Esta ley establece que los ciudadanos tienen derecho a interactuar por medios electrónicos con las administraciones públicas y que las administraciones públicas están obligadas a satisfacer estos derechos.

5.2.2. Características físicas

El DNI-e es una tarjeta de policarbonato plástico del tamaño de una tarjeta de crédito según el estándar ISO 7816 (85,6 x 53,98 mm). En el anverso de la tarjeta se encuentran los datos de filiación (nombre, apellidos, sexo, fecha de nacimiento,...), una fotografía del titular en blanco y negro, el número de DNI del ciudadano y fecha de expedición. En el reverso aparece el lugar de nacimiento, nombre de los padres, domicilio,...El DNI-e no contiene otras

informaciones personales (sanitaria, fiscal, tráfico,...) impresas o en el chip.

En el documento del DNI-e se usan las siguientes técnicas de impresión de seguridad para evitar falsificaciones:

- Guilloches: son dibujos complejos con muchos detalles basados en motivos geométricos y normalmente formados por líneas continuas entrelazadas. Son difíciles de recrear o reproducir.

- Impresión en iris: proceso de coloración que consigue un cambio gradual del color y protege de la copia.

- Tintas invisibles: tintas que contienen sustancias fluorescentes que sólo son visibles bajo luz ultravioleta.

- Microtextos: son líneas o motivos compuestos por letras o números de tamaño muy reducido, apenas perceptibles a simple vista.

- Imagen codificada: mediante herramientas informáticas especiales determinados datos, como el nombre del titular, se embeben en la fotografía del titular. Esta información es invisible para el ojo humano.

- OVI (tinta ópticamente variable): tinta con pigmentos ópticamente variables que presenta grandes variaciones de color en función del ángulo de observación o de iluminación.

- Kinegrama: es una estructura microscópica de difracción similar a un holograma. No es tridimensional, sino que al moverla muestra animaciones gráficas.

- Grabado por láser: imágenes o texto se consiguen quemando capas de policarbonato sensibles al láser.

- Grabado en relieve: en el anverso de la tarjeta se graba un motivo apreciable al tacto.

Estas y otras medidas de seguridad que se aplican a documentos oficiales pueden consultarse en la siguiente página web del Consejo de la UE: *http://www.consilium.europa.eu/prado/ES/glossaryPopup.html*.

5.2.3. Características electrónicas

El propósito de la parte electrónica del DNI-e es dotar al documento de las capacidades criptográficas necesarias para permitir al titular acreditar su identidad electrónica.

Actualmente el DNI-e usa el chip *ST19WL34* fabricado por la empresa ST-Microelectronics (*http:// www.st.com*) que se dedica al diseño y producción de circuitos integrados. El chip está formado por un microprocesador de 8 bits, 6 KBytes de RAM, 224 KBytes de ROM para el almacenamiento del sistema operativo y programas y 34 Kbytes de EEPROM para datos de usuario. La memoria EEPROM es capaz de retener datos durante 10 años y soporta 500.000 ciclos de borrado-escritura. Entre las características criptográficas del chip cabe destacar que contiene un acelerador hardware mejorado para DES y una librería de soporte para algoritmos simétricos (DES y TripleDES), una librería software para AES-128 y un procesador aritmético modular de 1.088 bits con librería de soporte para algoritmos asimétricos (multiplicación modular rápida y elevación al cuadrado con el método de Montgomery). Además incluye un generador de números aleatorios FIPS 140 nivel 2, un módulo para el cálculo de funciones CRC, tres timers de 8 bits, reloj interno y bus interno de interconexión. Las especificaciones del chip pueden encontrarse en *http://www.dnielectronico.es/PDFs/st19wl34.pdf*.

El sistema operativo que gobierna el chip se denomina *DNIe v1.1* y ha sido desarrollado por la *Fábrica Nacional de Moneda y Timbre - Real Casa de la Moneda (FNMT-RCM)* a partir de las especificaciones de la Dirección General de la Policía y la Guardia Civil. Posteriormente fue certificado según el estándar *Common Criteria (CC)*.

Debido a que la tecnología está en constante evolución y para afrontar posibles contingencias relacionadas con el suministro y fabricación de los chips se prevé contar pronto con un segundo chip de otro fabricante.

5.2.4. Contenido del chip

La información de la memoria EEPROM del chip está distribuida en tres zonas con diferentes niveles y condiciones de acceso. Sólo es posible realizar operaciones de lectura en las tres zonas; el ciudadano no puede escribir datos en el chip. La primera zona, denominada pública, es accesible sin restricciones. La segunda zona, denominada privada, es accesible por el ciudadano mediante el PIN. Finalmente, la tercera zona, denominada de seguridad, sólo es

accesible por el ciudadano en los puntos de actualización del DNI-e ubicados en las comisarías.

- Zona pública:
  - Certificado CA intermedia emisora.
  - Claves Diffie-Hellman.
  - Certificado X.509 de componente.
- Zona privada:
  - Certificado de Firma (No Repudio).
  - Certificado de Autenticación (Digital Signature).
- Zona de seguridad:
  - Datos de filiación del ciudadano (los del soporte físico).
  - Imagen de la fotografía.
  - Imagen de la firma manuscrita.

En el chip se almacenan las claves criptográficas y algunos datos de gestión.

- Datos criptográficos:
  - Clave RSA pública de autenticación (2048 bits).
  - Clave RSA pública de no repudio (2048 bits).
  - Clave RSA privada de autenticación (2048 bits).
  - Clave RSA privada de firma (2048 bits).
  - Clave pública de la CA raíz (4096 bits).
  - Claves *Diffie-Hellman*.
- Datos de gestión:
  - Traza de fabricación.
  - Número de serie del soporte.

En el área de seguridad se almacenan algunos datos biométricos del titular para hacer posible la identificación del ciudadano mediante la lectura de los datos y posterior comparación con los almacenados en la tarjeta. Los datos biométricos almacenados son las características de las huellas dactilares del ciudadano correspondientes a sus dos dedos índice siempre que sea posible.

La verificación de la identidad del ciudadano a través de las huellas dactilares se lleva a cabo mediante un lector de huellas que captura la huella dactilar del ciudadano y la procesa para obtener sus puntos característicos, posteriormente se realiza la comparación con la información biométrica almacenada en la tarjeta. Por seguridad, esta comparación se realiza en el interior de la tarjeta y se denomina *match-on-card (MoC)*. El acceso a los datos biométricos contenidos en el DNI-e se restringe a las aplicaciones e instalaciones de la Dirección General de la Policía y de la Guardia Civil para los siguientes usos:

- Desbloqueo del PIN.
- Renovación de claves y certificados.
- Verificación de identidad en los puestos de gestión.

El DNI-e contiene dos certificados digitales del ciudadano usados para autenticación y firma electrónica respectivamente. Existe un certificado adicional denominado *Certificado de Componente* emitido para autenticar el propio chip y cifrar la comunicación con él. Las parejas de claves correspondientes a cada certificado (clave pública y privada) se generan internamente mediante el generador de números aleatorios interno, en presencia del ciudadano y se marcan como no exportables para garantizar que sólo existirá una única copia de cada clave privada y que ésta residirá siempre en el interior del chip. Cualquier operación criptográfica que requiera el uso de las claves privadas debe ejecutarse en el interior del chip ya que no se permite bajo ningún concepto que la clave privada abandone el chip. El chip protege las claves privadas y, además, actúa como un coprocesador criptográfico si se le suministra el PIN para probar que somos los usuarios legítimos de la tarjeta. Las claves públicas se envían tras su generación a la Autoridad de Certificación para su inclusión en los correspondientes certificados digitales. La clave pública se exporta de la tarjeta almacenada en un certificado *Card Verificable* firmado por una clave de autenticación propia de la tarjeta. Este certificado *Card Verificable* es enviado a la PKI del DNI-e formando parte de una solicitud de certificación en

formato PKIX-CMP. Una vez emitidos los certificados se incorporan a la tarjeta para ser utilizados en operaciones posteriores. Los certificados digitales del titular contenidos en el chip pueden ser extraídos del chip para su proceso.

5.2.5. Perfiles de los certificados

Los *Certificados de Identidad Pública* vinculan la identidad de una persona física (nombre, apellidos y número del Documento Nacional de Identidad) a una determinada clave pública para garantizar la autenticidad, integridad y no repudio. Toda esta información está firmada electrónicamente por la Autoridad de Certificación creada al efecto.

Los Certificados de Identidad Pública emitidos por el sistema del DNI-e serán conformes a las siguientes normas:

- RFC 3280: Internet X.509 Public Key Infrastructure - Certificate and CRL Profile, Abril 2002.
- ITU-T Recommendation X.509 (2005): Information Technology - Open Systems Interconnection - The Directory: Authentication Framework.
- ETSI TS 101 862 V1.3.1 (2004-03): Qualified Certificate Profile, 2004.
- RFC 3739: Internet X.509 Public Key Infrastructure - Qualified Certificate Profile, Marzo 2004.

Cada ciudadano dispone de dos Certificados de Identidad Pública emitidos por la Dirección General de la Policía (Ministerio del Interior) y que tienen como finalidad:

- *Certificado de Autenticación*: garantizar electrónicamente la identidad del ciudadano.
- *Certificado de Firma*: permitir la firma electrónica reconocida de documentos.

El Certificado de Autenticación asegura que la comunicación electrónica se realiza con la persona titular del DNI pero no demuestra voluntad de firma. Por tanto su uso se restringe a aquellas operaciones que requieren confirmar la identidad y acceso seguro a sistemas remotos. La Ley de Firma Electrónica recoge la posibilidad de usar este certificado para acreditar la identidad del

ciudadano para la expedición de certificados reconocidos por parte de otros Prestadores de Servicios de Certificación sin necesidad de la presencia del ciudadano. El cuadro 5.1 representa la estructura y contenido del Certificado de Autenticación según el *Documento de Políticas y Prácticas de Certificación*.

Cuadro 5.1: Certificado de Autenticación

Certificado de Autenticación de Ciudadano		
Campo	Contenido	C
Campos de X509v1		
1. Versión	v3	
2. Serial Number	No secuencial	
3. Signature Algorithm	SHA256withRSAEncryption SHA1withRSAEncryption	
4. Issuer Distinguished Name	CN=AC DNIE XXX; OU=DNIE; O=Dirección General de la Policía; C=ES	
5. Validez	30 meses	
6. Subject	CN=APELLIDO1 APELLIDO2, NOMBRE; G=NOMBRE; SN=APELLIDO1; NUMERO DE SERIE=NIF; C=ES	
7. Subject Public Key Info	Algoritmo=RSA Encryption; Longitud clave=2048 bits	
Campos de X509v2		
1. IssuerUniqueIdentifier	No se utiliza	
2. SubjectUniqueIdentifier	No se utiliza	
Extensiones de X509v3		
1. Subject Key Identifier	Derivada de usar la función de hash SHA-1 sobre clave pública del sujeto	No
2. Authority Key Identifier	Derivada de usar la función de hash SHA-1 sobre clave pública de la CA emisora	No

Certificado de Autenticación de Ciudadano		
Campo	Contenido	C
3. KeyUsage		Si
DigitalSignature	1	
ContentCommitment	0	
Key Encipherment	0	
Data Encipherment	0	
Key Agreement	0	
Key Certificate Signature	0	
CRL Signature	0	
4. extKeyUsage	No se utiliza	
5. privateKeyUsagePeriod	No se utiliza	
6. CertificatePolicies		No
Policy Identifier	2.16.724.1.2.2.2.4	
URL DPC	http://www.dnie.es/dpc	
Notice Reference		
7. Policy Mappings		
8. Subject Alternate Names	No se utiliza	No
9. Issuer Alternate Names	No se utiliza	No
10. Subject Directory Attributes	dateOfBirth	
11. Basic Constraints		Si
Subject Type	Entidad Final	
Path Length Constraint	No se utiliza	
12. Policy Constraints	No se utiliza	
13. CRL Distribution Points	No se utiliza	No
14. Auth. Information Access	OCSP: http://ocsp.dnie.es; CA: http://www.dnie.es/certs/ACraiz.crt	No
15. netscapeCertType	No se utiliza	

Certificado de Autenticación de Ciudadano		
Campo	Contenido	C
16. netscapeRevocationURL	No procede	
17. netscapeCAPolicyURL	No procede	
18. netscapeComment	No procede	
19. Biometricinfo	Hash de los datos biométricos SHA256/SHA1	No
20. personalDataInfo (2.16.724.1.2.2.3.1)	Hash de los datos biográficos (impresos en el DNI-e) SHA1/SHA256	
21. qcstatements	id-etsi-qcs-QcCompliance id-etsi-qcs-QcSSCD	

El Certificado de Firma se utiliza para la firma de documentos garantizando la integridad del documento y el no repudio de origen. Es un certificado X.509 v3 que tiene activo en el *Key Usage* el bit de *ContentCommitment* (no repudio). Este certificado convierte la firma electrónica avanzada en firma electrónica reconocida, permitiendo su equiparación legal con la firma manuscrita. El cuadro 5.2 representa la estructura y contenido del Certificado de Firma según el *Documento de Políticas y Prácticas de Certificación*.

Cuadro 5.2: Certificado de Firma de Ciudadano

Certificado de Firma de Ciudadano		
Campo	Contenido	C
Campos de X509v1		
1. Versión	v3	
2. Serial Number	No secuencial	
3. Signature Algorithm	SHA256withRSAEncryption SHA1withRSAEncryption	
4. Issuer Distinguished Name	CN=AC DNIE XXX; OU=DNIE; O=Dirección General de la Policía; C=ES	

Certificado de Firma de Ciudadano		
Campo	Contenido	C
5. Validez	30 meses	
6. Subject	CN=APELLIDO1 APELLIDO2, NOMBRE; G=NOMBRE; SN=APELLIDO1; NUMERO DE SERIE=NIF; C=ES	
7. Subject Public Key Info	Algoritmo=RSA Encryption; Longitud clave=2048 bits	
Campos de X509v2		
1. IssuerUniqueIdentifier	No se utiliza	
2. SubjectUniqueIdentifier	No se utiliza	
Extensiones de X509v3		
1. Subject Key Identifier	Derivada de usar la función de hash SHA-1 sobre clave pública del sujeto	No
2. Authority Key Identifier	Derivada de usar la función de hash SHA-1 sobre clave pública de la CA emisora	No
3. KeyUsage		Si
DigitalSignature	0	
ContentCommitment	1	
Key Encipherment	0	
Data Encipherment	0	
Key Agreement	0	
Key Certificate Signature	0	
CRL Signature	0	
4. extKeyUsage	No se utiliza	
5. privateKeyUsagePeriod	No se utiliza	
6. CertificatePolicies		No
Policy Identifier	2.16.724.1.2.2.2.3	
URL DPC	http://www.dnie.es/dpc	

Certificado de Firma de Ciudadano		
Campo	Contenido	C
Notice Reference		
7. Policy Mappings		
8. Subject Alternate Names	No se utiliza	No
9. Issuer Alternate Names	No se utiliza	No
10. Subject Directory Attributes	dateOfBirth	
11. Basic Constraints		Si
Subject Type	Entidad Final	
Path Length Constraint	No se utiliza	
12. Policy Constraints	No se utiliza	
13. CRL Distribution Points	No se utiliza	No
14. Auth. Information Access	OCSP: http://ocsp.dnie.es; CA: http://www.dnie.es/certs/ACraiz.crt	No
15. netscapeCertType	No se utiliza	
16. netscapeRevocationURL	No procede	
17. netscapeCAPolicyURL	No procede	
18. netscapeComment	No procede	
19. Biometricinfo	Hash de los datos biométricos SHA256/SHA1	No
20. personalDataInfo (2.16.724.1.2.2.3.1)	Hash de los datos biográficos (impresos en el DNI-e) SHA1/SHA256	
21. qcstatements	id-etsi-qcs-QcCompliance id-etsi-qcs-QcSSCD	

En el chip hay un tercer certificado denominado *Certificado de Componente*. Su propósito es establecer un canal autenticado y cifrado entre la tarjeta

y sus drivers mediante un protocolo de autenticación mutua definido en CWA 14890. Este certificado no está accesible por los interfaces estándar PKCS #11 y CSP.

El DNI-e tiene un rango privado de OIDs por el cual todas las extensiones propietarias comienzan con el prefijo 2.16.724.1.2.2.3. En la actualidad hay definida una extensión propietaria denominada *PersonalDataInfo* con el OID 2.16.724.1.2.2.3.1 que contiene el hash de los datos biográficos (datos impresos en el DNI-e).

Los nombres contenidos en los certificados se basan en el formato X.500. El DN (*Distinguished Name*) para los certificados de ciudadano se compone de los siguientes elementos: CN (Common Name), GN (Givename), SN (Surname), SerialNumber y C (Country). El SerialNumber es el número de DNI con letra.

5.2.6. PKI del DNI electrónico

El OID de la DPC (*Declaración de Prácticas y Políticas de Certificación*) es 2.16.724.1.2.2.2.1 a los que se añaden 2 dígitos más con el formato X.Y correspondientes a la versión y release respectivamente.

La estructura del DNI-e soporta y utiliza CRLs X.509 versión 2 (v2). Las CRLs emitidas por el sistema DNI-e son conformes a las normas RFC 3280 y ITU-T Recommendation X.509 (2005). La PKI del DNI-e no publica CRLs en repositorios de acceso libre. Las CRLs sólo están disponibles como medio para intercambiar información de estado de los certificados con los Prestadores de Servicios de Validación. DNI-e publicará una nueva CRL en el momento en que se produzca cualquier revocación. El documento DPC también dedica una sección a los certificados de OCSP responder.

El DNI-e contribuirá a la aparición de empresas prestadoras de servicios de valor añadido (sistemas de cifrado, sellos de tiempo, ...) y prestadores de servicios de certificación a los ciudadanos privados ya que el DNI-e es un medio suficiente para acreditar la identidad y datos personales de los interesados. Puede ser utilizado como medio de identificación para la realización de registros fuertes sin que las empresas se vean obligadas a realizar una inversión grande en el despliegue y mantenimiento de una estructura de registro.

Las renovaciones de los certificados se realizan con cambio de claves. La tarjeta de soporte físico tiene un periodo de validez, a contar desde el momento de la expedición o renovación de:

- Cinco años: cuando el titular no haya cumplido los 30 años en el momento de la expedición o renovación.
- Diez años: cuando el titular haya cumplido los 30 años y no haya alcanzado los 70.
- Permanente: para mayores de 70 años y mayores de 30 años con la condición de gran inválido.
- Por un año: en casos excepcionales (cuando el ciudadano no pueda aportar la documentación requerida).

Los certificados electrónicos incorporados al DNI-e tendrán un período de vigencia de 30 meses siempre que este período no supere el del soporte físico, en cuyo caso la fecha de caducidad del certificado vendrá determinada por la del soporte. Todas estas consideraciones implican unos escenarios de renovación de certificados que se detallan en la DPC.

Según la DPC la revocación de un certificado es el acto por el cual se deja sin efecto la validez de un certificado antes de su fecha de caducidad. La revocación de un certificado conlleva la pérdida de validez del mismo e inhabilita el uso legítimo del mismo por parte del titular. No se contempla la revocación individual de uno de los certificados del DNI-e, sino que se revocarán simultáneamente los dos certificados. La revocación de un certificado tendrá como consecuencia la comunicación a terceros que dicho certificado ha sido revocado, siempre que se solicite la verificación del mismo a través de uno de los prestadores de servicios de validación. La revocación de un certificado obliga, en todos los casos, a la presencia física del titular. La DGP (Dirección General de Policía) puede solicitar de oficio la revocación de un certificado si tuviera conocimiento o sospecha del compromiso de la clave privada del ciudadano o cualquier otro hecho que recomendara emprender dicha acción. Una vez se ha validado la solicitud de revocación, ésta se lleva a cabo inmediatamente. En ningún caso el tratamiento de la revocación puede superar las 24 horas. La verificación de las revocaciones es obligatoria para cada uso de los certificados de identidad pública. El procedimiento ordinario de verificación de la validez de un certificado será la consulta a los prestadores de servicio de validación que, a su vez, indicarán el estado del certificado mediante el protocolo OCSP.

No se contempla la suspensión de certificados.

La tarjeta soporte del DNI-e es un dispositivo seguro de creación de firma certificado. Las claves privadas se generan dentro de la tarjeta y no pueden

ser exportadas en ningún caso. No se efectúa, por tanto, archivo de la clave privada de los certificados.

Autoridades de Certificación

La *Dirección General de Policía* (Ministerio del Interior) actúa como Autoridad de Certificación (CA) relacionando dos pares de claves a través de sendos certificados conformes con la DPC.

Las Autoridades de Certificación que componen la PKI del DNI-e son:

- *AC raíz*: Autoridad de Certificación de primer nivel. Esta CA sólo emite certificados para sí misma y sus CA subordinadas. Únicamente estará en funcionamiento durante la realización de las operaciones para las que se establece. Su *Nombre Distinguido* es: CN = AC RAIZ DNIE, OU = DNIE, O = DIRECCION GENERAL DE LA POLICIA, C = ES. Tiene un periodo de validez de 30 años (desde febrero del 2006 hasta febrero del 2036).

- *AC Subordinadas*: Autoridades de Certificación subordinadas de *AC raíz*. Su misión es la emisión de certificados para titulares de DNI-e. Su periodo de validez es de 15 años (hasta 2021). En la actualidad (2011) la PKI del DNI-e consta de 3 ACs Subordinadas denominadas:

  - CN = AC DNIE 001, OU = DNIE, O = DIRECCION GENERAL DE LA POLICIA, C = ES.
  - CN = AC DNIE 002, OU = DNIE, O = DIRECCION GENERAL DE LA POLICIA, C = ES.
  - CN = AC DNIE 003, OU = DNIE, O = DIRECCION GENERAL DE LA POLICIA, C = ES.

La incorporación o cese de operación de una CA son causa de la modificación del documento de la DPC.

Autoridades de Registro

La *Autoridad de Registro* está constituida por todas las oficinas de expedición del Documento Nacional de Identidad, y tienen por misión realizar las funciones de asistencia a la Autoridad de Certificación en los procedimientos

y trámites relacionados con los ciudadanos para su identificación, registro y autenticación y de esta forma garantizar la asignación de claves al solicitante. La situación geográfica serán las *Oficinas de Documentación* de la *Dirección General de la Policía* y las instalaciones habilitadas para los equipos móviles, en aquellos lugares donde no existe *Comisaría de Policía*, así como otros lugares que a tal efecto determine el órgano encargado de la expedición y gestión del DNI-e.

No se va a detallar aquí el proceso de expedición, pero sí destacar que una de las novedades más importantes en la expedición del DNI-e frente al DNI tradicional es la modificación del proceso para realizarse en un único acto administrativo permitiendo que el ciudadano obtenga su nuevo documento con una sóla visita al centro de expedición.

Autoridades de Validación

Las *Autoridades de Validación* tienen como función la comprobación del estado de los certificados emitidos por DNI-e. La información sobre los certificados electrónicos revocados se almacena en las denominadas listas de revocación de certificados (CRLs). Existen diversos modos de consulta de estas listas, pero en el caso del DNI-e se ha optado por un único protocolo de consulta basado en el estándar OCSP (*Online Certificate Status Protocol*).

Cuando una aplicación requiere determinar el estado actual de un certificado compone una petición OCSP y la envía mediante el protocolo HTTP a la URL del servicio de validadición contenida en el atributo AIA (*Authority Information Access*) de cada certificado. El valor actual del mencionado atributo es:

- *OCSP*, http://ocsp.dnie.es.
- *CA*, http://www.dnie.es/certs/ACraiz.crt.

La petición contiene datos del certificado sobre el que se realiza la consulta. En concreto la CA emisora y el número de serie del certificado con objeto de identificar unívocamente el certificado. El servicio de validación (OCSP Responder) realiza el acceso a las CRLs y averigua si dicho certificado está incluido en ellas. Posteriormente compone una respuesta con el estado, fecha y motivo que se firma con un certificado de *Autoridad de Validación* para mantener la integridad de la respuesta y como prueba de que la CA ha autorizado a la *Autoridad de Validación* a realizar dicha función de comprobación.

Según el documento DPC, la validez de los certificados de la Autoridad de Validación no será superior a los 6 meses. Además, tal y como contempla el estándar RFC 2560, la CA emisora incluirá en el certificado del OCSP Responder la extensión *id-pkix-ocsp-nocheck* para indicar que los clientes OCSP deben confiar en el prestador de servicios de validación durante el período de vida del certificado asociado. No obstante, no se descarta incluir en el futuro la extensión AIA en los certificados de OCSP Responder para comprobar la validez de dichos certificados.

Este servicio de consulta debe prestarse tal y como establece la Ley 59/2003 de firma electrónica, en su artículo 18 apartado d: garantizando *la disponibilidad de un servicio de consulta rápido y seguro*. La Dirección General de la Policía como *Prestador de Servicios de Certificación*, delega la función de comprobación del estado de revocación de los certificados DNI-e emitidos en varias entidades públicas externas que actúan como Autoridades de Validación. Se ha optado por asignar las funciones de Autoridad de Validación a entidades diferentes de la Autoridad de Certificación a fin de aislar la comprobación de la vigencia de un certificado de los datos de identidad de su titular para reforzar, aún más si cabe, la transparencia del sistema.

Hoy en día existen dos Autoridades de Validación (que cumplen con los objetivos de universalidad y redundancia):

- *Ministerio de la Presidencia*, que prestaría los servicios de validación al conjunto de las Administraciones Públicas.
- *Fábrica Nacional de Moneda y Timbre - Real Casa de la Moneda*, que prestaría sus servicios de validación con carácter universal: ciudadanos, empresas y Administraciones Públicas.

No se descarta que en el futuro la Dirección General de la Policía autorice a otros organismos a prestar servicios de Autoridad de Validación orientados a segmentos de usuarios concretos.

5.2.7. Uso del DNI-e

Para usar el DNI-e se necesita contar con elementos hardware y software que permitan a una aplicación que utilice certificados digitales comunicarse con el chip de la tarjeta y utilizar los certificados y claves contenidos en él.

Para operar con el DNI-e se requiere un elemento hardware denominado *lector de tarjetas inteligentes* (Smart Card Reader) con conexión a nuestro

equipo informático (PC, tablet, smartphone,...). Existen multitud de lectores de tarjetas inteligentes que se conectan de formas muy variadas a nuestro equipo. La forma más habitual es a través de un puerto USB, aunque también puede usarse un puerto serie RS232 o una interfaz PCMCIA.

El sistema operativo de nuestro equipo debe poder reconocer tanto el lector como el chip de la tarjeta del DNI-e por lo que se necesita instalar 2 componentes software. Por un lado el driver del lector y, por otro, el módulo criptográfico de la tarjeta. Para reconocer el lector hay que instalar un driver que nos suministrará el fabricante o bien emplear, si lo soporta, un driver universal creado por Microsoft y denominado CCID que existe en la mayoría de los sistemas operativos. El módulo criptográfico del DNI-e está disponible para Windows, Mac OS X, Sun Solaris 10 y algunas distribuciones del sistema operativo Linux. El módulo criptográfico del DNI-e es una librería que invocan las aplicaciones para realizar operaciones criptográficas sin trasladar a la aplicación la complejidad de gestionar diversos tipos de gestores y tarjetas. Hay actualmente dos especificaciones de módulo criptográfico. La primera especificación se basa en la arquitectura del sistema operativo Windows y se denomina *CryptoAPI*. La segunda se denomina *Cryptoki* y se usa principalmente en entornos basados en sistema operativos diferentes de Microsoft Windows. En los sistemas operativos Windows, algunas aplicaciones usan el interfaz CryptoAPI y otras, como por ejemplo el navegador Mozilla Firefox emplean el interfaz Cryptoki. Los fabricantes de tarjetas suministran ambos módulos con el fin de facilitar que la mayoría de las aplicaciones puedan operar con la tarjeta. El DNI-e es un recurso local por lo que es imprescindible que el código de programa que se comunica con él se ejecute en el equipo local. Las aplicaciones de escritorio, dado que se ejecutan en su totalidad en la máquina local del usuario no tienen habitualmente problemas para el acceso a las funciones del DNI-e. En las aplicaciones cliente-servidor los componentes que hacen uso del DNI-e deben ser ejecutados en el equipo donde se encuentra físicamente la tarjeta. Las aplicaciones web se ejecutan principalmente en el servidor y envían al navegador el resultado de su ejecución en lenguaje HTML para su presentación. Para solventar la restricción de ejecución local de código que accede al DNI-e se hace uso de los mecanismos de ejecución local de código que proporcionan los diferentes navegadores, como son los *Applets* de Java o los componentes *ActiveX*. La tecnología Java es multiplataforma, mientras que la tecnolgía ActiveX es específica de equipos con sistemas operativos Windows.

5.3. Tendencias

5.3.1. Criptografía

A medida que la criptografía evoluciona las PKIs deben adaptarse para incorporar las innovaciones y algoritmos que surgen en ese campo.

En el año 2001 NIST seleccionó el algoritmo Rijndael como el nuevo Advanced Encryption Standard (AES) para sustituir DES y Triple-DES como algoritmos estándares de cifrado simétrico. Esto ha conllevado en los últimos años la progresiva implantación de este algoritmo en las PKIs de los distintos fabricantes y la progresiva desaparición de DES y Triple-DES por el hecho de que se consideran algoritmos poco seguros. AES proporciona cifrado simétrico usando claves de 128 bits, 192 bits y 256 bits. Hoy en día todos los fabricantes de PKIs soportan AES (Entrust, Verisign, Microsoft,...).

La seguridad de un sistema viene dada por la seguridad del algoritmo más débil que utiliza, por ello la incorporación de AES en las PKIs implica también una revisión de la seguridad del resto de algoritmos. Por ejemplo, AES ha provocado la publicación por parte del NIST de tres nuevas funciones hash (SHA-256, SHA-394 y SHA-512) y aumentos en las longitudes de claves asimétricas usadas en los algoritmos de cifrado o firma digital.

Actualmente las mejoras criptográficas afectan básicamente dos aspectos de las PKIs: funciones hash e implementación de algoritmos basados en curvas elípticas.

En la actualidad el NIST está liderando un concurso en busca de funciones hash de tercera generación (SHA-3). Concretamente ahora está realizando una serie de conferencias para seleccionar los algoritmos candidatos y la proclamación del vencedor está prevista para el segundo trimestre del año 2012. Por tanto en el futuro inmediato habrá novedades en el campo de las funciones hash.

Tras la publicación de AES se comenzó a trabajar en algoritmos de clave pública que ofrecieran una seguridad equivalente. El número de bits de las claves asimétricas aumenta con el número de bits de la clave simétrica de AES. Para AES-128 la longitud de una clave pública que ofrece una seguridad similar usando los algoritmos RSA o Diffie-Hellman es de 3200 bits, pero la longitud de las claves públicas equivalentes para AES-192 y AES-256 está más allá de la capacidad de las smart cards o tokens USB actuales. AES-192 requiere una clave pública RSA de 7500 bits y AES-256 requiere una

clave pública RSA de 15000 bits. El problema no es el almacenamiento de las claves sino la generación del par de claves y el tiempo necesario para realizar las operaciones con la clave privada. La criptografía de curvas elípticas reduce considerablemente el número de bits de la clave y facilita las operaciones por lo que es previsible en el futuro inmediato el incremento de las smart cards y tokens USB que soporten criptografía de curvas elípticas. Previendo esta tendencia Microsoft también ha incorporado la criptografía de curvas elípticas en Windows Vista y Windows 2008.

También hay algo de investigación para conseguir que RSA se adapte a la capacidad de las smart cards. Multi-Prime RSA permite usar más de dos factores primos y ello provoca que las operaciones no consuman tantos recursos de procesador. No obstante, la tendencia actual predominante es hacia la criptografía de curvas elípticas.

5.3.2. Tendencias arquitectónicas

En la actualidad todas las arquitecturas PKI que se han comentado en el capítulo de diseño y despliegue se han implementado y se dispone de una buena experiencia de cada una de ellas.

En las grandes organizaciones se ha observado una tendencia progresiva a disminuir el número de CAs. Inicialmente se distribuían CAs en los distintos departamentos pero se observó que el coste de mantenimiento era demasiado elevado y por ello en la actualidad las CAs se centralizan. Si las CAs se diseñan adecuadamente esta tendencia es perfectamente asumible. Como contrapartida el número de RAs para el proceso de registro aumenta. Esta separación de tareas es positiva ya que permite a la organización concentrarse en un número reducido de CAs y dotarlas de una infraestructura mejor.

La arquitectura de CA puente se ha impuesto para proporcionar interoperabilidad entre PKIs, pero debido a la dificultad al comparar los documentos CP se tiende a uniformizar dichos documentos. La tendencia actual es que el número de CA puentes aumente aunque ello pueda llevar a complejos algoritmos de construcción de caminos de certificación.

5.3.3. Certificados

Los certificados X.509v3 han permanecido relativamente estables durante los últimos años y así se prevé que sigan. Debido a la flexibilidad del meca-

nismo de las extensiones, no hay ninguna razón que obligue a modificar el formato de los certificados. La tendencia actual es hacia la creación de nuevas extensiones a medida que se detecten nuevas necesidades. Otros formatos de certificados como SPKI no han logrado imponerse. Los certificados X.509v3, aunque siguen teniendo detractores, se benefician de años de experiencia y desarrollo. No hay expectativas de que los certificados X.509v3 sean reemplazados.

Los nuevos desarrollos en el campo de los certificados se centran en los *Certificados de Atributos* (AC) y en los certificados *proxy*.

Una aplicación de los certificados podría ser el control de acceso a recursos. La información de autorización de acceso podría ubicarse en una extensión de los certificados de clave pública. Sin embargo, esta estrategia no es buena por dos motivos. En primer lugar la información de acceso se actualiza más rápidamente que la información de un certificado de clave pública, esto obligaría a revocar y reemitir el certificado cada vez que la información de acceso se renovase. Esto sería caro. En segundo lugar, la CA que emite los certificados de clave pública probablemente no conoce la información de acceso.

Los Certificados de Atributos (AC) asocian al poseedor del certificado con información de control de acceso (atributos). Un certificado AC no contiene una clave pública por lo que el certificado de clave pública se usa en primer lugar para autenticar y, a continuación, el AC asocia atributos con la identidad autenticada. Los ACs pueden ser de larga duración (semanas o meses) o de corta duración (unas horas). Si son de corta duración el emisor de ACs no necesita mantener información de revocación ya que el certificado expirará antes de que la información de revocación sea compilada y distribuida. Si la duración de los ACs es de semanas o meses es necesario mantener la información de revocación. Los ACs se definen en los documentos X.509 y RFC5755. Los ACs se han intentado usar para delegar autorizaciones pero con poco éxito hasta la fecha por su complejidad. Los ACs originan infraestructuras denominadas PMIs (Privilege Management Infrastructures) que constituyen hoy en día un campo de importante actividad.

Los certificados proxy están definidos en el documento RFC3820. Son un tipo de certificado X.509v3 que permite delegar autorizaciones. Por ejemplo Alice puede delegar en Bob un subconjunto de sus autorizaciones. Los certificados proxy son emitidos por certificados de usuarios o bien por otros certificados proxy y utilizan la extensión crítica *Proxy Certificate Information*.

Otro tipo de certificados cuyo uso se está extendiendo son los certificados CV (Card Verifiable). Este tipo de certificados son los que se usan en las infraestructuras del e-pasaporte. Este tipo de certificados tienen periodos de validez cortos y no se usan esquemas de revocación. Por consiguiente, los componentes de una infraestructura de certificados CV deben coordinarse para la gestión eficiente de solicitudes de certificados frecuentes. Los estándares del e-pasaporte fueron establecidos por ICAO (International Civil Aviation Organization) y son seguidos por todos los países que los implementan, entre ellos España.

5.3.4. CRLs, OCSP y SCVP

Cuando surgieron las PKIs pocas revisaban el estado de los certificados. Poco a poco se fue implementando el mecanismo de las CRLs para validar el estado de los certificados. Pero las CRLs emitidas por algunas CAs eran muy grandes.

El resultado fue un creciente interés en la validación de certificados online. Así surgió OCSP y se espera que las PKIs evolucionen e incluyan este tipo de mecanismos de validación de certificados online y que incluso reemplacen a las CRLs. La tendencia actual es producir CRLs todavía, si bien no son consultadas directamente por las entidades finales que intentan validar un certificado. Las CRLs proporcionaran información a los servidores OCSP.

El IETF ha completado la especificación del protocolo SCVP (Server-based Certificate Validation Protocol). SCVP permite a las aplicaciones PKI delegar la construcción y validación de caminos de certificados. En el futuro SCVP puede reemplazar OCSP porque el camino completo de certificados puede ser validado con una única solicitud.

5.3.5. Biometría

El uso de la biometría con técnicas criptográficas es un campo prometedor de investigación. Se han sugerido varios métodos para asegurar una clave privada con datos biométricos. Particularmente, la combinación de biometría y smart cards es un campo con un prometedor futuro a corto plazo. La biometría se considera esencial en el diseño de un sistema de identificación seguro.

Un sistema biométrico consta de dos etapas: alistamiento y validación. Durante la fase de alistamiento se adquiere una muestra biométrica (voz, cara,

huella dactilar, iris scan,...). La muestra se procesa y se genera un patrón biométrico a partir de ella. Las sucesivas validaciones del usuario comparan patrones biométricos. Fruto de la comparación se emite una puntuación. Los diseñadores del sistema establecen un umbral en la puntuación para aceptar o rechazar la comparación. El umbral se define en función de criterios de seguridad y comodidad.

Un sistema de identificación basado en la biometría permite la verificación de la información de identidad impresa o almacenada en la tarjeta basada en información biométrica en lugar de información que se sabe, como sería el caso de un PIN. Esto incrementa la seguridad del sistema y mejora la exactitud y rapidez de la identificación del poseedor de la tarjeta. Con la muestra biométrica almacenada en la smart card la comparación puede hacerse en la propia tarjeta inteligente. Las últimas smart cards tienen suficiente potencia de cálculo y memoria para realizar la comparación en la propia lógica de la tarjeta, en lugar de en un lector externo. En la actualidad ya se fabrican smart cards que incorporan lectores de huellas dactilares.

5.4. Viabilidad futura de las PKIs

La criptografía asimétrica ofrece un pequeño pero importante conjunto de servicios que la criptografía simétrica no puede proveer. Además no se vislumbran alternativas a las PKIs para gestionar las claves privadas y públicas. Puede que en el futuro la infraestructura sufra modificaciones y difiera de la actual, pero siempre habrá una infraestructura de la que no se podrá prescindir. De todo esto se desprende la viabilidad futura de las PKIs.

Como ya se ha mencionado anteriormente una PKI es una infraestructura y no un fin en sí misma. Los principales usos que se han dado a la infraestructura PKI son: soportar aplicaciones de negocio, proporcionar autenticación fuerte a PMIs y soportar requisitos legales. Hay una cierta evolución en el marketing de las PKIs. Inicialmente se intentó vender la tecnología con poco éxito, a continuación se pasó a vender los servicios proporcionados por la tecnología (autenticación, firmas digitales,...). En el futuro se promoverán los servicios pero en términos de los tipos de aplicaciones que cada servicio soporta mejor.

La externalización puede desempeñar un papel muy importante en la expansión de las PKIs por dos motivos básicos: costes y experiencia. El coste

de una PKI es alto y cualquier posibilidad de ahorro es bienvenida. En ocasiones es difícil tratar con problemas que se presentan por primera vez en una instalación, no obstante pueden ser comunes a muchas instalaciones, de ahí la importancia de la experiencia. Sin embargo, la externalización no es la panacea ya que también presenta sus inconvenientes como, por ejemplo, la integración cuando la PKI se extiende a varias organizaciones.

Las PKIs han sido mayoritariamente usadas cuando se tenían que proteger transacciones de alto valor debido al alto coste de las PKIs. No obstante, se debe entender mejor el tipo de transacciones a las que una PKI puede proporcionar protección para que la industria comprenda mejor cuándo la aplicación de PKIs es necesaria. Microsoft está introduciendo en sus productos (Windows 2003, Windows 2008, Windows Vista,...) la criptografía de clave pública. Esto pone las PKIs al alcance de muchas pequeñas y medianas empresas. Las PKIs pueden ser utilizadas para pequeñas aplicaciones internas con un mínimo impacto. Comprender el tipo de escenarios en los que una forma simplificada de PKI es aplicable constituye una importante vía de expansión de las PKIs.

Un obstáculo para la difusión de las PKIs es la redacción de los documentos CPs y CPSs. La solución estriba en disponer de mejores ejemplos de estos documentos adaptados a áreas de negocio específicas o tipos de modelos de procesos. El estándar RFC3647 proporciona un punto de partida para la redacción de los documentos pero es demasiado vago para los que abordan la tarea por primera vez. Es necesario disponer de modelos de los documentos CPs y CPSs más adaptados al mundo de la empresa.

En definitiva la viabilidad futura de las PKIs pasa por comprender mejor cómo usar los servicios proporcionados por una PKI:

- Autenticación. La capacidad de proporcionar autenticación fuerte es vista hoy en día como uno de los principales beneficios de una PKI. Una aplicación destacada consiste en soportar mecanismos de autorización.

- Integridad. La capacidad de las firmas digitales de proporcionar un mecanismo de integridad es también muy importante. Este mecanismo se usa en no pocas aplicaciones.

- No-repudio. La capacidad de proporcionar no-repudio es una faceta destacada. Desgraciadamente dificultades legales hacen que su aplica-

ción sea compleja. Esta complicaciones deben ser eliminadas para que se pueda usar realmente en la práctica.

El éxito vaticinado inicialmente por los precursores de las PKIs dista mucho de haberse cumplido. Las causas que han causado este fracaso son diversas pero entre ellas sobresale la confusión sobre lo que la tecnología podía conseguir. En muchos entornos la motivación para el despliegue era errónea o inexistente y la adaptación de aplicaciones resultaba muy compleja.

Sin embargo no sería justo menospreciar los logros conseguidos. En la actualidad hay diversas motivaciones que promueven la paulatina implantacion de PKIs. Por un lado existen iniciativas gubernamentales; por otro, algunas aplicaciones de Internet favorecen el uso de PKIs. La posibilidad de firmar documentos junto con la integración de las PKIs con otras tecnologías (Web Services, formularios electrónicos, PMIs,...) constituyen un fuerte impulso para la implantación de PKIs.

Actualmente, por ejemplo, muchos gobiernos se hallan en alguna etapa del despliegue de una PKI entre sus ciudadanos. El certificado emitido por la autoridad gubernamental puede usarse para acceder a servicios on-line del propio gobierno o como pasaporte electrónico al ser reconocido por las PKIs de otros países, posiblemente a través de una cross-certificación, o para identificarnos en otras PKIs y obtener un nuevo certificado del mismo modo que el DNI nos identifica y permite obtener otros documentos certificados.

Internet se está convirtiendo en un repositorio de información altamente confidencial como, por ejemplo, datos financieros personales o relativos a la salud. El tratamiento de estos datos requiere una sólida identificación de los usuarios que los manipulan. Las firmas digitales usadas para firmar transacciones y almacenarlas también constituyen un estímulo para la implantación de PKIs. Parece inevitable que el crecimiento de Internet conllevará un aumento de la demanda de los servicios que una PKI es capaz de ofrecer.

En la actualidad se dedica una atención especial a comprender las necesidades de los programadores de aplicaciones y otros elementos del entorno donde se instalará la PKI para garantizar una implantación tan efectiva y no traumática como sea posible. Se han diseñado protocolos orientados a aligerar las tareas que se deben llevar a cabo en el cliente PKI (OCSP, SCVP, DSV,...). Si se consigue consensuar formatos y detalles de los protocolos el cliente PKI se puede simplificar mucho, hacerlo resistente a cambios en el proveedor de la PKI y evitar actualizaciones de software en los clientes.

Predecir si las PKIs sobrevivirán y prosperarán es difícil. Lo que es un heho es que la tecnología PKI está en continua evolución para adaptarse mejor al entorno y ha madurado de forma considerable en los últimos años. Es una tecnología que ha ido ganando aceptación y se ha extendido poco a poco a pesar de las dificultades.

Apéndice A

Información adicional

A.1. Introducción

En este apéndice se incluyen algunas cuestiones relativas a precios de smart cards, tokens criptográficos y lectores de tarjetas. Asimismo se comentan precios de obtención y renovación de certificados. También se incluye información sobre aplicaciones que soportan PKIs (navegadores, clientes de correo electrónico, generadores de PDF,...) y aplicaciones relacionadas con el DNI-e. Por último se comenta dónde obtener ejemplos de los documentos CP y CPS.

A.2. Smart cards y tokens USB

Esta sección está dedicada a la tecnología de las smart cards. Básicamente se comenta información de fabricantes y precios. Este tipo de tecnología usa un chip o circuito integrado que puede contener un microcontrolador con memoria o simplemente memoria. La smart card se conecta con un lector físicamente o a través de RF (radiofrecuencia). Las smart cards que disponen de microcontrolador tienen la capacidad de almacenar grandes cantidades de datos y realizar funciones como, por ejemplo, cifrar, autenticar, comparar patrones biométricos e interactuar inteligentemente con un lector de smart cards. Esta tecnología se ajusta a estándares internacionales y está disponible en una gran variedad de formatos (tarjetas de crédito, SIMs de teléfonos móviles, e-passports y tokens USB).

Gemalto (http://www.gemalto.com) es uno de los fabricantes más desta-

cados de hardware criptográfico. Ofrece smart cards criptográficas al precio de 2 €en su gama de productos *TPC Classic*. Este tipo de tarjetas son compatibles con la CSP API y con PKCS #11 permitiendo su uso en cualquier tipo de aplicación PKI. Dispone de un amplio rango de algoritmos criptográficos y el modelo *Classic TPC IM CC* es completamente compatible con la normativa europea de firmas digitales. Además de firmas digitales, el soporte de criptografía de clave pública incluye generación de claves en la propia tarjeta y descifrado de claves de sesión. Soporta claves RSA de hasta 2048 bits. Utilizan la tecnología *JavaCard* que es una plataforma diseñada por Sun que permite a las smart cards ejecutar aplicaciones basadas en Java. Otro grupo de smart cards que vende Gemalto es la familia .NET que ejecuta una versión adaptada de la plataforma .NET y está especialmente orientada al entorno Windows. Los sistemas Windows soportan nativamente este tipo de smart cards por lo que no es necesario instalar ningún componente software. Este tipo de tarjetas permite una solución sencilla y con un coste razonable en la implantación de una autenticación de 2 factores. Su coste es de unos 20 €. Gemalto también comercializa lectores de smart cards. Hay diferentes modelos de lectores para las diferentes interfases del PC que soportan cualquier tipo de smart card. El precio de los lectores USB es de unos 25 €; el de los lectores PCMCIA es de unos 12 €; los lectores a través de puerto serie RS232 cuestan unos 34 €; también hay tarjetas *ExpressCard 54* al precio de unos 50 €; finalmente, los lectores *Contactless* por puerto USB oscilan alrededor de los 70 €. Otro tipo de productos íntimamente relacionados con los anteriores son los *Tokens USB*. Este tipo de dispositivos se conectan al PC a través de un conector USB y portan en su interior una smart card; en definitiva, integran una tarjeta inteligente y un lector. Además de proporcionar autenticación de 2 factores dificultan enormemente la manipulación indebida de la smart card. El resultado combina robustez, portabilidad, comodidad y seguridad. Un precio orientativo podría rondar los 20 €.

La empresa española *C3PO* (http://www.c3po.es) , si bien no fabrica smart cards propiamente, sí las distribuye. Concretamente comercializa una tarjeta criptográfica con el nombre de *Criptonita* al precio de unos 12 €. El modelo de CPU es *ST19XL34V2*, dispone de una memoria ROM de 96 Kbytes, 4 Kbytes de RAM y 34 Kbytes de EEPROM. Soporta RSA, 3DES y SHA-1. Sus principales aplicaciones son el almacenamiento y gestión de certificados digitales X.509v3, correo seguro, conexión segura cliente servidor y Windows logon. Por contra, C3PO fabrica lectores de smart cards en todas las

variedades de conectividad (RS232, USB, PCMCIA, Express Card), externos, internos e integrados en teclado. El rango de precio de los lectores oscila entre los 13 €de los modelos USB y los 82 €para un dispositivo de hasta 4 tarjetas chip gestionado mediante un puerto serie. La empresa europea Kalysis (http://www.kalysis.com) con sede social en España comercializa tokens USB y lectores de smart cards.

Cuando se usa en combinación con la biometría, una smart card deviene más personal y privada. La biometría proporciona una asociación única entre el poseedor de la tarjeta y los datos almacenados en ella de manera que sólo la persona correcta puede acceder a los datos. Los últimos microprocesadores usados en smart cards aportan suficiente potencia de cálculo como para llevar a cabo el reconocimiento del patrón biométrico en la propia smart card. Esto proporciona un sistema de identificación más seguro todavía y capaz de proporcionar autenticación de hasta 3 factores. La empresa Novacard (http://www.novacard.de) presentó en el año 2004 la primera smart card con un sensor de huella dactilar que cumplía los estándares ISO. Hoy en día dicha tarjeta es capaz de reconocer el patrón biométrico en el propio chip de la tarjeta. En la actualidad se intenta conseguir suficiente potencia de cálculo en las smart cards para que, además de reconocer patrones biométricos, sean capaces de generar dichos patrones y almacenarlos en la propia smart card.

A.3. Certificados

Son muchas las organizaciones que emiten certificados como parte de su negocio. En esta sección se examinan brevemente los principales tipos de certificados, sus precios y funciones. Las aplicaciones que más habitualmente usan certificados son SSL y la firma de código.

Son muchas las empresas que emiten certificados SSL y es imposible enumerarlas todas por lo que simplemente se hará mención de las más destacadas. SSL es el protocolo de seguridad más ampliamente desplegado en la actualidad. Esencialmente proporciona un canal seguro entre dos máquinas que operan en Internet o algún otro tipo de red. Típicamente se usa cuando se requiere una conexión segura entre un navegador y un servidor web. Básicamente hay 2 tipos de certificados SSL. Por un lado, algunas organizaciones requieren simplemente certificados SSL para confidencialidad. En tal caso, la CA, antes de emitir el certificado, verifica el derecho del solicitante a usar un dominio

específico junto con algunas validaciones adicionales sobre la organización. Por otro lado, otras organizaciones requieren que se mejore la confianza en su seguridad e identidad. En tal caso, la CA, antes de emitir el certificado, comprueba el derecho del solicitante a usar un dominio específico, pero además realiza un examen profundo de la organización. Este tipo de certificados se denomina *certificados EV*. Un certificado EV es un certificado de clave pública X.509v3 emitido de acuerdo con un conjunto específico de criterios de verificación de la identidad. El proceso de emisión de estos certificados está estrictamente definido en las *EV Guidelines* y fue formalmente ratificado en el *CA/Browser Forum* del año 2007. Los pasos requeridos para la emisión de un certificado EV por parte de una CA incluyen:

- Verificar la existencia legal, física y operativa de la entidad.
- Verificar que la identidad de la entidad coincide con la de los registros oficiales.
- Verificar que la entidad tiene el derecho exclusivo a usar el dominio nombrado en el certificado EV.
- Verificar que la entidad ha autorizado la emisión del certificado EV.

Un segundo conjunto de directrices, *EV Audit Guidelines*, establece los criterios bajo los cuales una CA debe ser auditada anualmente para poder emitir certificados EV. Una de las claves del éxito del protocolo SSL es que no requiere la intervención del usuario; todo sucede de un modo automático. Las últimas versiones de los navegadores más populares soportan los certificados EV. Los navegadores suelen mostrar un candado cerrado y la cadena de texto *https* en lugar de *http* en la barra de direcciones URL para demostrar que están usando el protocolo SSL. Si además el fondo de la barra de direcciones URL es de color verde y aparece el nombre de la compañía legalmente propietaria de la web, entonces se está usando un certificado EV.

Verisign (http://www.verisign.com) vende certificados SSL de 4 tipos. Los más caros son certificados EV que cuestan 1.499 $ por año y usan claves para cifrar de entre 128 y 256 bits. Los más baratos son certificados convencionales que cuestan 399 $ por año y usan claves entre 40 y 256 bits [1].

[1] Estas longitudes se refieren a claves secretas y pueden deberse a puro marketing. No tiene sentido que un certificado restrinja el uso de criptografía de clave secreta.

GlobalSign (http://www.globalsign.com) también comercializa diferentes categorías de certificados SSL que oscilan entre los 899 $ por año para certificados EV y unos 174 $ por año para certificados SSL convencionales. Entrust (http://www.entrust.com) emite certificados EV a unos 400 $ anuales, mientras que un certificado estándar ronda los 190 $ por año. La *Agència Catalana de Certificació* también emite certificados SSL al precio de unos 43 €a partir de 6 unidades. En general, los precios de las renovaciones de certificados son muy similares al de las emisiones propiamente dichas. Izenpe, la empresa vasca de certificación, vende certificados SSL convencionales y también EV. La FNMT-RCM a través del proyecto CERES dispone en su catálogo de certificados SSL.

En cuanto a la firma de código, cabe decir que el código no firmado puede ser alterado de manera que incluya spyware o malware; por ello se recomienda a los usuarios finales que no ejecuten código no firmado. Una vez se ha firmado el código los usuarios están seguros de la identidad del vendedor o fabricante y de que el software no ha sido alterado desde que fue publicado. Diferentes plataformas de software tienen diferentes requisitos y diferentes herramientas para firmar código. Verisign ofrece certificados para la firma de código para:

- Microsoft Authenticode: permite firmar .exe, .dll, .cab, .ocx, .msi, ... al precio de unos 500 $ anuales.
- Sun Java: permite firmar .jar para sobremesa y plataformas móviles al precio de unos 500 $ anuales.
- Microsoft Office y VBA: permite firmar macros VBA de Microsoft Office al precio de unos 500 $ anuales.
- Adobe AIR: permite firmar .air o .airi al precio de unos 500 $ anuales.
- Macromedia Shockwave: firma ficheros creados con Macromedia Director 8 Shockwave Studio al precio de unos 500 $ anuales.
- Authentic IDs for BREW al precio de unos 400 $.

Los precios de las renovaciones son los mismos.

GlobalSign ofrece certificados para firmar código en las plataformas siguientes: Microsoft Authenticode, Java, Adobe AIR, VBA de Microsoft Office, Apple y objetos Mozilla y Netscape al precio de unos 230 $ anuales.

La empresa vasca Izenpe y la FNMT-RCM ofrecen también certificados para firmar código.

A.4. Aplicaciones

Entre los principales fabricantes de plataformas PKI se encuentran Entrust (http://www.entrust.com), Globalsign (http://www.globalsign.com) y Verisign (http://www.verisign.com). Entrust y Verisign ofrecen plataformas PKI ubicadas en las instalaciones del cliente. No obstante, debido a los altos costes de este tipo de PKIs se tiende a soluciones externalizadas. Las soluciones en casa del cliente se reducen a organizaciones gubernamentales interesadas en fomentar la administración electrónica, documentos de identidad o e-pasaportes y grandes corporaciones con necesidades de seguridad muy críticas. Los tres fabricantes ofrecen soluciones externalizadas en las que las CAs se hallan fuera de las instalaciones del cliente. Esta es la tendencia actual para pequeñas y medianas organizaciones.

La empresa española Safelayer (http://www.safelayer.com) ofrece una completa gama de productos para el desarrollo de soluciones avanzadas de Infraestructura de Clave Pública incluyendo la generación y gestión de certificados digitales, la validación de certificados digitales y el sellado electrónico de tiempo. Estos productos están principalmente orientados a organizaciones de tamaño mediano o grande. Safelayer proporciona componentes de su PKI en proyectos como el pasaporte electrónico y el DNI electrónico.

El protocolo HTTPS, desarrollado en el estándar RFC2818, capacita a los navegadores y servidores web para establecer conexiones TCP cifradas sobre TLS (*Transport Layer Security*) de manera que los datos no puedan ser interceptados o modificados. El protocolo TLS es el sucesor del protocolo SSL (*Secure Sockets Layer*). Para establecer la identidad de las entidades finales el servidor presenta un certificado de clave pública al navegador que es validado contra un conjunto de certificados firmados por unas cuantas autoridades de certificación comerciales que vienen preinstalados en el navegador o en el sistema operativo. El navegador puede, opcionalmente, presentar un certificado firmado al servidor para identificarse frente a éste. Esta posibilidad es raramente usada en la práctica y el cliente permanece anónimo hasta que se autentifica mediante algún otro mecanismo a nivel HTTP. Las nuevas versiones de los navegadores más populares (Opera, Firefox, Chrome, Internet

Explorer) implementan el protocolo OCSP para verificar que el certificado no ha sido revocado. Debido a la forma en que servidores web y navegadores manejan los certificados el usuario está obligado a confiar en que los certificados fueron correctamente preinstalados y en que las CAs son confiables y emiten certificados a webs legítimas. Todo esto hace que la situación diste mucho de ser óptima. Firefox, Opera y Chrome son multiplataforma, mientras que Internet Explorer sólo funciona bajo Windows.

En general, los sistemas de correo electrónico seguro siguen un esquema mixto en el que usan simultáneamente algoritmos simétricos y asimétricos para obtener el máximo beneficio de cada uno de estos tipos de cifrado: la velocidad de los sistemas simétricos junto con la seguridad en la gestión de claves de los sistemas de clave pública. Para hacer llegar un mensaje confidencial, se ha de enviar junto con el texto del mensaje, procesado mediante un algoritmo de cifrado simétrico, la clave secreta usada. Esta clave se cifrará con la clave pública del receptor del mensaje para que sólo éste pueda descifrar la clave y, con ella, el resto del mensaje. Este procedimiento se conoce con el nombre de sobre digital. Para probar la autenticidad del mensaje es necesario incluir una firma digital obtenida con un algoritmo de clave asimétrica a partir de la clave privada del emisor.

Los principales clientes de correo electrónico actuales (Outlook, Mozilla Thunderbird, Lotus Notes, ...) soportan la especificación S/MIME (Secure / Multipurpose Internet Mail Extensions) que usa certificados digitales basados en la norma X.509. S/MIME combina algoritmos criptográficos seguros con los estándares más difundidos de correo electrónico y ha sido pensado para conseguir la máxima interoperabilidad de modo que dos clientes de fabricantes diferentes no tengan problemas para establecer una comunicación segura. S/MIME, por otro lado, no es adecuado para usarse con clientes webmail ya que, debido a que la clave privada debe mantenerse localmente, complica la principal ventaja ofrecida por estos clientes que es la posibilidad de acceder el correo desde cualquier lugar. El cifrado requiere disponer del certificado del destinatario, lo cual es automático si antes se ha recibido un mensaje firmado por el destinatario. Algunos clientes de correo como Lotus Notes soportan OCSP, CRLs y smart cards o tokens USB. Outlook se basa, como Internet Explorer, en CryptoAPI y por ello puede soportar CRLs y OCSP. Notes y Thunderbird son multiplataforma mientras que Outlook sólo funciona bajo Windows.

Otro tipo de aplicaciones que usa las PKI son las VPN (*Virtual Private*

Network). Una VPN es una red privada de datos que hace uso de una infraestructura de telecomunicaciones pública como Internet. El uso de una VPN se puede contrastar con el de un sistema de líneas dedicadas usadas por una única compañía aunque el coste es mucho menor. Las tecnologías más comúnmente usadas por las VPNs son IPsec y SSL 3.0 o TLS. Cisco VPN Client permite establecer túneles cifrados extremo a extremo con Cisco Easy VPN servers; además es multiplataforma (Windows, Linux, Solaris, MAC OS X) y soporta SCEP. Otros fabricantes de software VPN son Check Point, SafeNet y Sonicwall.

Debido a su popularidad se comenta someramente el sistema de cifrado usado por Microsoft Office, si bien no utiliza propiamente criptografía de clave pública. El proceso que usa Microsoft Office para cifrar y descifrar sus ficheros no ha cambiado en esencia desde sus primeras versiones. El procedimiento comienza concatenando la password y una semilla de 16 bytes elegida aleatoriamente. Esta cadena de bits actúa como entrada de una función hash. La función hash se itera en repetidas ocasiones con su propia salida. Se seleccionan los primeros bits (entre 40 y 256 según la versión) como clave con la cual se cifran los datos. Por otro lado, se selecciona aleatoriamente una cadena de 128 bits llamada *Verifier* que se cifra con la clave y se almacena en el fichero Office (*EncryptedVerifier*). El Verifier se pasa una vez por la función hash, se cifra con la clave y se almacena en el fichero (*EncryptedVerifierHash*). Para verificar la password es necesario generar la clave, descifrar el Verifier que fue almacenado en el fichero, pasar el Verifier por la función hash, cifrar la cadena resultante y comparar el resultado con el EncryptedVerifierHash que también fue almacenado en el fichero. Este esquema ha ido cambiando con el tiempo la función hash y el algoritmo de cifrado. Inicialmente, en Office 97 y 2000 se usaban MD5 y RC4. Debido a que las leyes norteamericanas prohibían la exportación de criptografía, no era posible usar claves de más de 40 bits fuera de los Estados Unidos. Esto hacía posibles los ataques por fuerza bruta. Cuando aparecieron las versiones de Office XP y 2003 se habían derogado las leyes de exportación y se había desarrollado CryptoAPI. La implantación de CryptoAPI permitió usar SHA1 como función hash, incrementar la longitud de las passwords de 16 a 255 símbolos y usar claves de hasta 128 bits en el algoritmo RC4. Parecía que por primera vez se disponía de auténtica protección en los ficheros de Office, pero Hongjun Wu de la universidad de Nanyang descubrió que Microsoft usaba incorrectamente el algoritmo RC4. En la versión Office 2007 RC4 fue sustituido por AES y la función hash se

itera 50.000 veces. AES es un cifrador de bloques de 128 bits, mientras que SHA1 genera 160 bits por lo que se necesitan 2 bloques. Sin parase a pensar, Microsoft rellenó los últimos 96 bits del segundo bloque con ceros. Esto facilita la ruptura del algoritmo, aunque todavía no ha sido posible. Office 2010, la última versión, realiza algunas mejoras relativas a la protección de ficheros con passwords como son el control de la complejidad de la password, soporte a la nueva API *Cryptograhy: Next Generation* (CNG) y algunos cambios en la interfase con el usuario que facilitan el uso y comprensión de esta característica del programa.

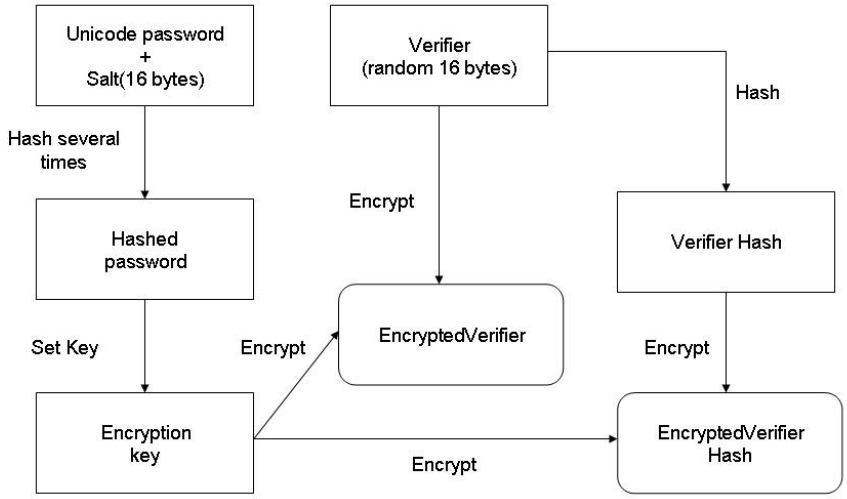

Figura A.1: Cifrado MS Office

Para preservar la integridad de los documentos, Microsoft Office 2010 utiliza la firma digital validada. La firma validada, además de incluir el certificado del firmante, añade información temporal procedente de una autoridad de tiempo fiable e información sobre la vigencia del certificado en el momento de la firma. Las firmas de Office son compatibles con el estándar W3C *XML Advanced Electronic Signatures* (XAdES). Este tipo de firmas siguen siendo legalmente válidas si el certificado correspondiente expira.

El paquete ofimático *OpenOffice.org* es una versión de código abierto de la suite *StarOffice* cuyo desarrollo esponsorizaban principalmente Sun Microsystems y Oracle Corporation (tras su adquisición de Sun en el 2010). En Abril de 2011 Oracle anunció que dejaría de soportar el desarrollo de OpenOffice.org y muchos de sus principales contribuidores trabajan en la actualidad en el proyecto similar *LibreOffice*. Está disponible para varios sistemas operativos (Windows, Mac OS, Linux,...). Del mismo modo que Microsoft Office, OpenOffice.org implementa las firmas digitales siguiendo las recomendaciones W3C XML DSIG, una de cuyas variantes para firmas avanzadas es XAdES.

Siguiendo con las firmas digitales, la empresa Adobe, en su especificación PDF, definió un formato de firma propietario basado en el estándar PKCS #7 incrustado en el propio documento. A finales del 2007, ISO aprobó el formato PDF versión 1.7 como el estándar ISO 32001. Más recientemente, el estándar PAdES (ETSI TS 102 778) perfila ISO 32001 con la finalidad de poder incluir firmas electrónicas avanzadas en los documentos PDF. Los productos de Adobe permiten incorporar la firma digital de forma sencilla en los documentos PDF. La validación es posible realizarla con las últimas versiones del visor Adobe Acrobat Reader. Existen también múltiples productos comerciales y librerías de desarrollo de código abierto que permiten realizar las mismas funciones de firma y validación en ficheros PDF. *TrustedX* es una plataforma de servicios web de la empresa Safelayer que aporta autenticación, autorización, firma electrónica y protección de datos en Arquitecturas Orientadas a Servicios (SOA). Concretamente soporta firma electrónica de documentos (XAdES, PDF, ...) y mensajería S/MIME. Son varias las empresas que ofrecen PDF Suites que permiten, entre otras cosas, la firma electrónica de documentos PDF. Entre ellas se pueden citar a Aloaha, novaPDF, PDF Suite y Nitro PDF. Como software gratuito se puede citar *iText* que es una librería que permite manipular y también firmar documentos PDF. *iSafePDF* es una aplicación de código abierto basada en la librería *iText* que permite cifrar y firmar documentos PDF.

Otra aplicación popular basada en la criptografía de clave pública es *Encrypting File System* (EFS) de la familia de sistemas operativos Windows. Las *Listas de Control de Acceso* tradicionalmente usadas para proteger los ficheros de accesos no autorizados pueden ser fácilmente soslayadas por un atacante. La solución más ampliamente aceptada consiste en almacenar los ficheros cifrados en el medio físico (discos, CDs, pen drives,...). La familia de siste-

mas operativos Windows incluye EFS en su sistema de archivos NTFS. EFS combina la criptografía de clave simétrica y la de clave pública. EFS funciona cifrando el fichero con una clave simétrica denominada *File Encryption Key* (FEK) que se genera aleatoriamente. El algoritmo de cifrado simétrico usado varía dependiendo de la versión y configuración del sistema operativo. En las últimas versiones (Windows 7 y Windows 2008) se usa principalmente AES. La clave FEK se cifra, a su vez, con la clave pública asociada al usuario que procesa el fichero y se almacena en el fichero junto con los datos cifrados. En versiones previas, como Windows 2000, se usaba al usuario Administrador como *Data Recovery Agent*. El Data Recovery Agent podía descifrar todos los ficheros cifrados EFS ya que la clave FEK también se almacenaba cifrada con la clave pública del Data Recovery Agent. Esto permitía que cualquiera que pudiera apropiarse de la cuenta del Administrador pudiera acceder a cualquier fichero cifrado. En Windows XP y versiones posteriores ya no se requiere un Data Recovery Agent y se disminuye el riesgo de acceso no autorizado a la información cifrada. Los ficheros se descifran automáticamente cuando son copiados a otro volumen formateado con otro sistema de ficheros, por ejemplo FAT32, y cuando se transmiten sobre la red usando el protocolo SMB/CIFS. En principio, EFS puede funcionar en ausencia de PKI ya que si no hay una CA disponible se generan certificados autofirmados en la propia máquina.

Hay muchas aplicaciones de varios fabricantes que protegen el sistema de ficheros aunque la mayoría trabajan a nivel de disco y no de carpeta o fichero. SafeGuard Easy de la empresa Sophos protege datos confidenciales en PCs con sistemas operativos Windows cifrando la totalidad del disco. SecureDoc de WinMagic también proporciona cifrado completo de discos en plataformas Windows, MAC y Linux. Symantec y Check Point ofrecen productos de similares características.

La firma de código permite firmar digitalmente ejecutables y scripts de modo que se garantiza su autor e integridad. Su uso está muy difundido en entornos distribuidos en los que el origen de un bloque de código puede no ser evidente, como ocurre con applets de Java y controles ActiveX. Otra utilidad muy importante es la distribución de actualizaciones y parches de software ya existente. Los servicios de actualización de la mayoría de las distribuiciones de Linux, Apple MAC OS X y Microsoft Windows usan la firma de código para asegurarse de que no es posible distribuir código malicioso a través de los parches de software. Los certificados usados pueden provenir de una

CA y deben poderse trazar hasta una autoridad raíz de confianza. Verisign, GlobalSign y TC TrustCenter son emisores de este tipo de certificados. Otra alternativa es que los certificados sean autofirmados. En este caso el usuario debe de obtener la clave pública directamente del desarrollador. El producto Lotus Notes de IBM ha usado la firma de código desde la primera versión. Otra aplicación habitual de la firma de código es JavaScript. Microsoft usa una firma de código basada en Authenticode en sus drivers. Puesto que los drivers se ejecutan en modo kernel, pueden desestabilizar el sistema o provocar agujeros en la seguridad. Por todo ello, los drivers son cuidadosamente validados y posteriormente firmados para certificar su seguridad. Los paquetes de desarrollo vienen con utilidades que permiten la firma de código.

A.5. Aplicaciones DNI-e

En esta sección se recogen algunas de las aplicaciones y servicios que podemos utilizar con el DNI-e. Si se desea ampliar esta información se puede consultar la web *http://zonatic.usatudni.es*.

Antes de poder operar con el DNI-e en nuestro ordenador, necesitamos disponer de un lector e instalar el módulo criptográfico adecuado a nuestro sistema operativo. La Dirección General de Policía y la Guardia Civil ofrece un servicio en la página web *http://www.dnielectronico.es* que permite verificar el correcto funcionamiento del DNI-e en nuestro ordenador. Tras conectarnos a la página web con un navegador soportado, debemos seguir las instrucciones y, si el conjunto funciona correctamente, se validará el certificado y se obtendrá una página con el contenido de los certificados digitales.

Las aplicaciones que usan el DNI-e se pueden clasificar del siguiente modo: firma de documentos, factura electrónica, autenticación de usuarios y otros tipos de aplicaciones. A continuación se comentarán brevemente estos tipos de aplicaciones.

Existen diversos productos, algunos gratuitos y otros con un coste reducido, que permiten realizar de forma sencilla la firma individual o de un conjunto reducido de documentos. Los formatos de firma básicos sólo incorporan información del documento y de los certificados de los firmantes. Los formatos básicos son *PDF Signature* para ficheros PDF, *XMLDSig* para ficheros XML y *PKCS #7/CMS* para ficheros binarios. Los formatos de firma avanzados incorporan en la firma otros elementos como el sellado de tiempo o

la respuesta OCSP de la autoridad de validación sobre la revocación del certificado. Estos formatos de firma facilitan el archivado y verificación de la firma de documentos a largo plazo (firma longeva). Los formatos avanzados son *CAdES* para ficheros binarios, *XAdES* para ficheros XML y *PAdES* para ficheros PDF.

eCoFirma es una aplicación multiplataforma y gratuita del Ministerio de Industria, Turismo y Comercio desarrollada en Java. Soporta la firma de cualquier tipo de fichero, la firma múltiple y tiene opciones para la verificación de firmas y para la realización de copias electrónicas de documentos. Usa el formato XAdES y encapsula los documentos firmados en un fichero único con la extensión *.xsig*. Se puede descargar de la web del Ministerio.

ESecure de la empresa KSI Seguridad Digital (http://www.ksitdigital.com) es una aplicación Windows de bajo coste que permite la firma, básica y avanzada, de cualquier fichero en formatos PDF, CMS y XML usando DNI-e u otras tarjetas como FNMT, Camerfirma, Izenpe,.... Además permite cifrar con contraseñas o certificados.

XolidoSign de Xolido (http://www.xolido.com) es una aplicación gratuita para Windows que permite la firma básica y avanzada de ficheros de cualquier tipo y su verificación. Permite usar distintos tipos de certificados, entre ellos el DNI-e. Permite la firma de múltiples documentos.

Otras aplicaciones de firma son *ClickSign* de isigma (http://www.isigma.es), *ProSign* de FirmaProfesional (http://www.firmaprofesional.com) y *ProFirma* de Albalia Interactiva (http://www.albalia.com).

También hay plataformas de firma electrónica avanzadas que se instalan en equipos servidores y permiten niveles elevados de rendimiento y alta disponibilidad. Entre ellas cabe citar *TrustedX* de Safelayer, *PSA* de CATCert, ZAIN de Izenpe, PortaSigma de isigma y @firma del Ministerio de Política Territorial y Administración Pública.

A pesar de que las aplicaciones de firma vistas pueden, en muchos casos, realizar las funciones para la generación y gestión de la factura electrónica hay una serie de aplicaciones y plataformas especializadas en esta tarea específica.

FACCIL (http://www.faccil.com) de Albalia Interactiva es un portal que permite el envío y recepción instantánea de facturas cumpliendo con la normativa legal. Las utilidades de FACCIL de conservación de facturas, legibilidad, control de accesos y firma electrónica garantizan la identidad del emisor de la factura y que la misma no ha sido alterada desde su emisión.

ecoFACTURA de KSI Seguridad Digital es una aplicación de escritorio

para la firma digital masiva de facturas y otras tareas relativas a la facturación: envío al cliente, archivado, reenvío de copias, ...

La *Digitalización Certificada* de facturas es el proceso que permite obtener copias digitales de las facturas con valor de original lo que hace posible destruir la factura en papel. Para cumplir los requisitos que lo permiten, la entidad que lleva a cabo la digitalización debe usar un software homologado por la AEAT (Agencia Estatal de Administración Tributaria) y realizar el proceso de acuerdo con un *Plan de Calidad*. El proceso implica firmas electrónicas avanzadas y admite varios formatos para las facturas digitalizadas (PDF, PNG, JPEG 2000,...). Hay una gran multitud de aplicaciones homologadas por la AEAT para este propósito. Se pueden consultar en la web http://www.aeat.es y, entre ellas, se pueden citar: TS-DIGCERT de T-Systems, INVESDOC DC de IECISA, I-FACT de Indra, FactUM de la Universidad de Murcia,... La lista es bastante extensa.

En cuanto a las aplicaciones de autenticación de usuarios cabe citar *IDOne Professional* de SmartAccess (http://www.smartaccess.es) que es una aplicación de pago que permite reemplazar o complementar el uso de contraseñas para acceder a ordenadores Windows con el DNI-e, otras tarjetas inteligentes y tokens USB, en ordenadores aislados o redes pequeñas. También SmartAccess dispone de otra aplicación para el acceso a ordenadores en grandes redes corporativas que cuenten con el Directorio Activo de Microsoft denominada *SmartID Corporate Logon*. Además del DNI-e permite usar otros certificados emitidos en smart card como los de la FNMT.

Finalmente, como ejemplos de otros tipos de aplicaciones se pueden citar el registro automático de visitantes con DNI-e y la firma de contratos. Algunos tipos de eventos, como ferias y congresos, precisan comprobar la identidad de muchas personas en poco tiempo y el DNI-e puede contribuir a mejorar estos procesos. Este tipo de aplicación puede extenderse al registro de clientes en hoteles, visitantes de un edificio público, ... La posibilidad de leer automáticamente los datos del chip (nombre, apellidos y número del DNI) permite agilizar el registro y la búsqueda en bases de datos además de evitar errores de transcripción. La validación de los certificados permite evitar el uso de tarjetas anuladas o robadas. A este efecto se puede usar el software *DNI-e Xplorer* de la empresa Smart Access que se instala en ordenadores con sistema operativo Windows. En cuanto a la firma de contratos, cabe citar el servicio online *http://www.tractis.com* que permite crear, negociar y firmar contratos e incluye herramientas para su redacción a través de una serie de

plantillas; aunque también permite la carga de documentos elaborados con un procesador de textos.

A.6. CP y CPS

En Internet se pueden conseguir muchos ejemplos de documentos de prácticas y políticas de certificación. En esta sección se expondrán algunos ejemplos.

El documento *Infraestructura de Clave Pública DNI electrónico* es la declaración de prácticas y políticas de certificación del DNI-e. Este documento se puede descargar de *http://www.dnielectronico.es*.

El documento *Declaración de Prácticas de Certificación* de Izenpe puede obtenerse en la siguiente página web *http://www.izenpe.com*.

La FNMT-RCM emite diferentes tipos de certificados y ha redactado varios documentos de prácticas de certificación. Todos ellos disponibles en *http://www.cert.fnmt.es*.

La empresa CATCert *Agència Catalana de Certificació* tiene sus documentos en la página web *http://www.catcert.cat*.

Verisign tiene su documento de prácticas de certificación en la página web *http://www.verisign.com/repository/cps*.

GlobalSign tiene varios documentos de prácticas de certificación en la página web *http://www.globalsign.com/repository*.

En definitiva, si se quiere saber dónde encontrar el documento de prácticas de certificación correspondiente a un certificado no hay más que consultar su extensión relativa a políticas del certificado.

Glosario

AC	Attribute Certificate
AES	Advanced Encryption Standard
AEAT	Agencia Estatal de Administración Tributaria
APEC	Asia-Pacific Economic Cooperation
ANSI	American National Standards Institute
API	Application Programming Interface
BSI	British Standards Institution
CA	Certification Authority
CAdES	CMS Advanced Electronic Signature
CAPI	Cryptographic API
CC	Common Criteria
CCID	Chip/Smart Card Interface Devices
CESG	Communications-Electronics Security Group
CMC	Certificate Management over CMS
CMP	Certificate Management Protocol
CNG	Cryptography Next Generation
CP	Certificate Policy
CPS	Certification Practice Statements
CRL	Certificate Revocation List
CRMF	Certificate Request Message Format
CV	Card Verifiable
CWA	CEN Workshop Agreement
DISP	Directory Information Shadowing Protocol
DN	Distinguished Name
DNI-e	Documento Nacional de Identidad Electrónico
DGP	Dirección General de Policía
DPC	Declaración de Prácticas y Políticas de Certificación
DPD	Delegated Path Discovery

DPV	Delegated Path Validation
ECDSA	Elliptic Curve Digital Signature Algorithm
EE	End Entity
EESSI	European Electronic Signature Standarization Initiative
EFS	Encrypting File System
ETSI	European Telecommunications Standards Institute
FEK	File Encryption Key
FIPS	Federal Information Processing Standards
FNMT-RCM	Fábrica Nacional de Moneda y Timbre - Real Casa de la Moneda
GCHQ	Government Communications Headquarters
GSS-API	Generic Security Service Application Program Interface
HSM	Hardware Security Module
HTTPS	Hypertext Transfer Protocol Secure (HTTPS)
ICAO	International Civil Aviation Organization
IEC	International Electrotechnical Commission
IETF	Internet Engineering Task Force
IPSec	Internet Protocol Security
ISO	International Organization for Standardization
IT	Information Technology
ITU	International Telecommunication Union
LDAP	Lightweight Directory Access Protocol
NIST	National Institute of Standards and Technology
NSA	National Security Agency
OID	Object Identifier
OVI	Optically Variable Ink
PAdES	PDF Advanced Electronic Signature
PDA	Personal Digital Assistant
PDF	Portable Document Format
PEM	Privacy Enhanced Mail
PGP	Pretty Good Privacy
PIN	Personal Identification Number
PKC	Public Key Cryptography
PKI	Public Key Infrastructure
PKIX	Public Key Infrastructure X.509
PMI	Privilege Management Infrastructure
RA	Registration Authority
RF	Radio Frequency

RFC	Request For Comments
ROI	Return On Investment
RSA	Rivest Shamir Adleman
RSA-PSS	Rivest Shamir Adleman Probabilistic Signature Scheme
RSA-OAEP	Rivest Shamir Adleman Optimal Asymmetric Encryption Padding
SCEP	Simple Certificate Enrollment Protocol
SCVP	Simple Certificate Validation Protocol
SET	Secure Electronic Transaction
SOA	Service Oriented Architecture
SP	Security Policy
SP	Special Publications
SPKI	Simple Public Key Infrastructure
SSL	Secure Sockets Layer
S/MIME	Secure Multipurpose Internet Mail Extensions
TCO	Total Cost of Ownership
TLS	Transport Layer Security
TS	Technical Specification
VPN	Virtual Private Network
W3C	World Wide Web Consortium
XAdES	XML Advanced Electronic Signature

Bibliografía

[1] W. Diffie, M. Hellman. *New Directions in Cryptography*. IEEE Transactions on Information Theory. 1976

http://www-ee.stanford.edu/~hellman/publications/24.pdf

[2] *History of Public-Key Cryptography*.

http://www.ics.uci.edu/~ics54/doc/security/pkhistory.html

[3] *Public Key Cryptography History*

http://www.livinginternet.com/i/is_crypt_pkc_inv.htm

[4] S. Singh. *The Code Book*

http://cryptome.org/ukpk-alt.htm

[5] B. Schneier, C. Ellison. *Ten risks of PKI: What you are not being told about Public Key Infrastructure*

http://www.schneier.com/paper-pki.html

[6] S. Lloyd. *Understanding Certification Path Construction*

http://www.oasis-pki.org/pdfs/Understanding_Path_construction-DS2.pdf

[7] S. Lloyd. *PKI Interoperability Framework*

http://www.oasis-pki.org/pdfs/PKIInteroperabilityFramework.pdf

[8] S. Lloyd. *CA-CA Interoperability*

http://www.oasis-pki.org/pdfs/ca-ca_interop.pdf

[9] S. Boeyen. *X.509 Profiles for various CA scenarios* 2004.

http://www.entrust.ca/resources/whitepapers.cfm

[10] G. Price. *PKI - An Insider's View (Extended Abstract)*, 2005.

http://www.rhul.ac.uk/mathematics/techreports

[11] C. McLaughlin. *Proposed Model for Outsourcing PKI*, 2008.

http://www.rhul.ac.uk/mathematics/techreports

[12] J. J. Nombela. *Firma Electrónica: Aspectos Técnicos de la Legislación y Aplicaciones: DNI-e*, 2005.

http://www.criptored.upm.es/guiateoria/gt_m213b.htm

[13] Verisign *Reducing Complexity and Total Cost of Ownership with Verisign Managed PKI*

http://www.verisign.com/authentication/information-center/authentication-resources/whitepaper-cost-effective-pki.pdf

[14] J. Crespo y otros *Hacia una nueva identificación electrónica del ciudadano: el DNI-e*

http://www.safelayer.com/pdf/DNIe.pdf

[15] M. Zalewski *Browser Security Handbook*

http://code.google.com/p/browsersec/wiki/Part2

[16] Microsoft *Security Overview for Office 2010*

http://technet.microsoft.com/en-us/library/cc179050.aspx

[17] Pavel Semjanov *MS Office encryption: keep making the old mistake*

http://www.password-crackers.com/blog/?p=48

[18] Pavel Semjanov *Encryption evolution in Microsoft Office*

http://www.password-crackers.com/blog/?p=16

[19] Hongjun Wu *The Misuse of RC4 in Microsoft Word and Excel*

http://www3.ntu.edu.sg/home/wuhj/research/msoffice/microsoft.pdf

[20] Microsoft *Introduction to Code Signing*

http://msdn.microsoft.com/en-us/library/ms537361.aspx

[21] Smart Card Alliance *Smart Cards and Biometrics*
http://www.smartcardalliance.org/resources/pdf/Smart_Cards_and_Biometrics_030111.p

[22] R. Housley, T. Polk. *Planning for PKI* Wiley, 2001.

[23] M. Benantar. *Introduction to the Public Key Infrastructure for the Internet* Prentice Hall, 2002.

[24] M. Balladelly, J. De Clercq. *Mission-Critical Active Directory* Digital Press, 2001.

[25] J. De Clercq. *Windows Server 2003 Security Infrastructures* Digital Press, 2004.

[26] H. Johner y otros. *Deploying a Public Key Infrastructure* IBM Redbooks, 2000.

[27] K. Schmeh. *Cryptography and Public Key Infrastructure on the Internet* Wiley, 2001.

[28] T. Austin. *PKI* Wiley, 2001.

[29] J. Buchmann. *Introduction to Cryptography* Springer, 2001.

[30] A. Nash y otros. *PKI: Implementing and Managing E-security* RSA Press, 2001.

[31] C. Adams, S. Lloyd. *Understanding PKI: Concepts, Standards and Deployment Considerations* 2nd Edition Addison-Wesley, 2002.

[32] A. Dent, C. Mitchell. *Users's Guide to Cryptography and Standards* Artech House, 2004.

[33] B. Komar. *Windows Server 2008 PKI and Certificate Security* Microsoft Press, 2008.

[34] G. Valiente. *Composición de textos científicos con Latex* Edicions UPC, 1997.

[35] S. Choudhury. *Public Key Infrastructure Implementation and Design* M&T Books, 2002.

[36] S. Turner, R. Housley. *Implementing E-mail Security and Tokens* Wiley, 2008.

[37] R. Sarwat *DNI-e Tecnología y Usos* Informática64, 2010.

[38] E. Barker, W. Barker, W. Burr, W. Polk, M. Smid. *NIST SP 800-57: Recommendation for Key Management - Part 1* 2007.

http://csrc.nist.gov/publications/nistpubs/800-57/sp800-57-Part1-revised2_Mar08-2007.pdf

[39] *Security Requirements for Cryptographyc Modules* 2001.

http://csrc.nist.gov/publications/fips/fips140-2/fips1402.pdf

[40] *Federal Public Key Infrastructure (PKI) X.509 Certificate and CRL Extensions Profile* 2005.

http://www.idmanagement.gov/fpkipa/documents/fpki_certificate_profile.pdf

[41] RFC 2247 *Using Domains in LDAP/X.500 Distinguished Names*

http://datatracker.ietf.org/doc/rfc2247

[42] RFC 2560 *Internet X.509 Public Key Infrastructure Online Certificate Status Protocol - OCSP*

http://datatracker.ietf.org/doc/rfc2560

[43] RFC 2818 *HTTP over TLS*

http://datatracker.ietf.org/doc/rfc2560

[44] RFC 2585 *Internet X.509 Public Key Infrastructure Operational Protocols: FTP and HTTP*

http://datatracker.ietf.org/doc/rfc2585

[45] RFC 2743 *Generic Security Service Application Program Interface Version 2, Update 1*

http://datatracker.ietf.org/doc/rfc2743

[46] RFC 2744 *Generic Security Service API Version 2: C-bindings*

http://datatracker.ietf.org/doc/rfc2744

[47] RFC 3647 *Internet X.509 Public Key Infrastructure Certificate Policy and Certification Practices Framework*

http://datatracker.ietf.org/doc/rfc3647

[48] RFC 3739 *Internet X.509 Public Key Infrastructure: Qualified Certificates Profile*

http://datatracker.ietf.org/doc/rfc3739

[49] RFC 3820 *Internet X.509 Public Key Infrastructure (PKI) Proxy Certificate Profile*

http://datatracker.ietf.org/doc/rfc3820

[50] RFC 4158 *Internet X.509 Public Key Infrastructure: Certification Path Building*

http://datatracker.ietf.org/doc/rfc4158

[51] RFC 4210 *Internet X.509 Public Key Infrastructure Certificate Management Protocol (CMP)*

http://datatracker.ietf.org/doc/rfc4210

[52] RFC 4211 *Internet X.509 Public Key Infrastructure Certificate Request Message Format (CRMF)*

http://datatracker.ietf.org/doc/rfc4211

[53] RFC 4398 *Storing Certificates in the Domain Name System (DNS)*

http://datatracker.ietf.org/doc/rfc4398

[54] RFC 4523 *Lightweight Directory Access Protocol (LDAP) Schema Definitions for X.509 Certificates*

http://datatracker.ietf.org/doc/rfc4523

[55] RFC 5055 *Server-Based Certificate Validation Protocol*

http://datatracker.ietf.org/doc/rfc5055

[56] RFC 5272 *Certificate Management over CMS (CMC)*

http://datatracker.ietf.org/doc/rfc5272

[57] RFC 5273 *Certificate Management over CMS (CMC): Transport Protocols*

http://datatracker.ietf.org/doc/rfc5273

[58] RFC 5274 *Certificate Management over CMS (CMC): Compliance Requirements*

http://datatracker.ietf.org/doc/rfc5274

[59] RFC 5280 *Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile*

http://datatracker.ietf.org/doc/rfc5280

[60] RFC 5652 *Cryptographic Message Syntax (CMS)*

http://datatracker.ietf.org/doc/rfc5652

[61] RFC 5653 *Generic Security Service API Version 2: Java bindings Update*

http://datatracker.ietf.org/doc/rfc5653

[62] RFC 5755 *An Internet Attribute Certificate Profile for Authorization*

http://datatracker.ietf.org/doc/rfc5755

[63] *ITU-T Recommendation X.509*

http://www.itu.int/rec/T-REC-X.509

[64] *PKCS #5: Password-Based Cryptography Standard*

http://www.rsa.com/rsalabs

[65] *PKCS #7: Cryptographic Message Syntax Standard*

http://www.rsa.com/rsalabs

[66] *PKCS #8: Private-Key Information Syntax Standard*

http://www.rsa.com/rsalabs

[67] *PKCS #10: Certification Request Syntax Standard*

http://www.rsa.com/rsalabs

[68] *PKCS #11: Cryptographic Token Interface Standard*
http://www.rsa.com/rsalabs

[69] *PKCS #12: Personal Information Exchange Syntax Standard*
http://www.rsa.com/rsalabs

[70] *Glosario del Consejo de la UE de seguridad de los documentos, medidas de seguridad y otros términos técnicos conexos*.
http://www.consilium.europa.eu/prado/ES/glossaryPopup.html

[71] *Proyecto de Declaración de Prácticas y Políticas de Certificación*
http://www.dnielectronico.es/PDFs/politicas_de_certificacion.pdf